Robustness of Multi-Objective Genetic Algorithm in the Optimization of Large Pipeline Networks

Kamal K. Botros, Ph.D., P.Eng.
NOVA Chemicals Corporation

Lunch Presentation to:
Canadian Operational Research Society
Calgary Section

September 19, 2007
### Published Papers:


Outline:

1. Motivation

2. Genetic Algorithm Methodologies

3. Example Application

4. Optimizing the Optimization

5. Conclusions
Motivation

Why Optimization in P/L Operation?

1. Reduce hydraulic analysis time through automation.
2. Improve operation.
3. Minimize fuel consumption.

Three basic objectives:

1. Minimum Fuel consumption.
3. Maximum (or desired) linepack.

1 MW @ 35% Th. Eff. @ $4/GJ

= $0.31 million/year of fuel & 4 ktonnes of CO2/year
Motivation

<table>
<thead>
<tr>
<th></th>
<th>With Electric</th>
<th>Without Electric</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alberta</td>
<td>928.4</td>
<td>928.4</td>
</tr>
<tr>
<td>Mainline</td>
<td>2345.9</td>
<td>2159.5</td>
</tr>
<tr>
<td>TQM</td>
<td>24.2</td>
<td>0</td>
</tr>
<tr>
<td>Foothills (incl BC)</td>
<td>374</td>
<td>354.2</td>
</tr>
<tr>
<td>Ventures</td>
<td>4.8</td>
<td>4.8</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>3677.3</strong></td>
<td><strong>3446.9</strong></td>
</tr>
</tbody>
</table>

- **Cost of Fuel = $1,068 million/year**
- **CO2 = 13,790 ktonnes/year**
- 1% saving = $10.7 million/year
- 1% saving = $137 ktonnes/year
Motivation

Maximum Throughput:

\[ Q_D = \sum_{k=1}^{N_D} Q_k , \quad Q_R = \sum_{k=1}^{N_R} Q_k \]

Maximum (or Desired) Linepack:

\[ LP = C \frac{LD^2 P_a}{Z_a T_a} \]
Motivation

3. Minimum Fuel Consumption:

\[ g(m, P_s, P_d) = \alpha \frac{mH}{\eta}, \forall (m, P_s, P_d) \in D \]

is subject to the following constraints:

1. \[
\begin{align*}
W_{r,\text{min}} &< W < W_{r,\text{max}} \\
N_{r,\text{min}} &< N_r < N_{r,\text{max}} \quad r \in R
\end{align*}
\]

\[
\text{Surge} < \left( \frac{Q}{N} \right)_r < \text{Stonewall}
\]

2. Multi-unit station operation and load sharing strategy

\[ P^L \leq P \leq P^U \]

3. Network Hydraulics: \[ A^T P^2 = \phi(\dot{m}) \]

\[ A\dot{m} = S \]
Optimization Methodology Selection

Primary Criteria:

- Must find global optimum independent of initialization

Secondary Criteria:

- Multi-objectives
- Robust
- Minimize solution time

Preferred Methodology: Genetic Algorithm (GA)

- Satisfies most criteria
General Formulation of Optimization Problem

\[
\min \{ f(x) : c_i(x) \leq 0, \ i \in I, \ c_i(x) = 0, \ i \in E \}
\]

Objective(s)

Inequality constraints

Equality constraints

Control Variables
(Decision Variable)
Classification of Optimization Method

1. **Gradient Based Methods.**

2. **Heuristic (Stochastic or Evolutionary) Methods**
Gradient-Based Optimization Chart

- **Continuous Unconstrained**
  - Linear or Non
    - Linear
      - Equality/Inequality
      - Bound Constrained
    - Quadratic
      - Equality/Inequality
      - Bound Constrained
    - Non-Linear
      - Bound Constraint
      - Equality/Inequality
      - Equality/Bound

- **Continuous Constrained**
  - Linear or Non

**Schemes**
- Newton’s Method
- SIMPLEX
- Interior Point
- Null-Space or Active Set Method
- Quasi-Newton
- Gradient Projection
  - Reduced Gradient Method
  - Sequential Linear Programming
  - Sequential Quadratic Programming
  - Augmented Lagrangian Method
Advantages and Disadvantages of Gradient-Based Optimization

**Advantages:**

- Less number of function calls.
- **Converges quickly** if the active set is near local minima.
- Accurately finds local minima.
- Less computer time.
- Numerous Constraint handling Techniques --> **flexibility in constraint handling**.

**Disadvantages:**

- Depends on the **Start Point**.
- Often trapped in **Local Minima** - Not suitable for finding global minima.
- First-order, second order derivatives, excessive matrix manipulation
- Not suitable for **large systems**.
- Not suitable for **multi-objective** optimization.
- Not suitable for **multidisciplinary** problem.
Fundamentals of Genetic Algorithms

- Algorithm based on natural evolution.
- Form of “survival of the fittest”.
- GA search from a population of points, not a single point.
- GA uses Fitness function, not the objective function itself.
- GA uses probabilistic transition rules, not deterministic rules.
- GA works with a coding of the parameter set, not the parameters themselves.
Genetic Algorithms

Advantages:

• Good for *global* optima and does not get trapped in local minima.
• **Gradient free.**
• Excellent for Multidisciplinary and **multi-objectives**.
• Does not care about *size of the system*.
• Suitable for **parallelization**.

Disadvantages:

• Large number of **function calls**.
• long computer **time** if parallelization is not used.
• May depend on internal **GA parameters** (crossover, mutation, ranking, selection, fitness function, and constraint handling).
How Genetic Algorithm Works!
Binary Encoding

Example: Pressure ($P_{\text{min}}=5000$ kPa, $P_{\text{max}}=6000$ kPa)

$11010$

Decoded Value = 26

\[
x_i = x_i^{\text{min}} + \frac{x_i^{\text{max}} - x_i^{\text{min}}}{2^l_i} DV(s_i)
\]

Pressure Value = \(5000 + (6000-5000) \times 26/2^5\)

= 5812.5 kPa
Fundamentals of Genetic Algorithms

Binary Encoding

Design Case (pipeline decision variables)

1011001011001010111001010101011001110

Q  Pd1  Pd2  N1  N2  V%  ST
Design Case Chromosome:
## Example of Decision Variable (Table 1 of 3)

<table>
<thead>
<tr>
<th>Description</th>
<th>Min</th>
<th>Max</th>
<th>Resolution</th>
<th>N</th>
<th>String Length (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proration Factor</td>
<td>900</td>
<td>1100</td>
<td>1</td>
<td>201</td>
<td>8</td>
</tr>
<tr>
<td>DPC at Saddle Hills CV (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>DPC at Gold Creek CV (kPa)</td>
<td>5865</td>
<td>7245</td>
<td>10</td>
<td>139</td>
<td>8</td>
</tr>
<tr>
<td>DPC at Knight CV (kPa)</td>
<td>4685</td>
<td>6175</td>
<td>10</td>
<td>150</td>
<td>8</td>
</tr>
<tr>
<td>Zama Lake #3 MS (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>DPC at Zama Lake CV (kPa)</td>
<td>3205</td>
<td>5655</td>
<td>10</td>
<td>246</td>
<td>8</td>
</tr>
<tr>
<td>DPC at Edson CV (kPa)</td>
<td>4005</td>
<td>6205</td>
<td>10</td>
<td>221</td>
<td>8</td>
</tr>
<tr>
<td>Meikle River Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Meikle River (kPa)</td>
<td>3200</td>
<td>4430</td>
<td>10</td>
<td>124</td>
<td>7</td>
</tr>
<tr>
<td>Valleyview Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Valleyview (kPa)</td>
<td>3200</td>
<td>4430</td>
<td>10</td>
<td>124</td>
<td>7</td>
</tr>
<tr>
<td>Gold Creek B Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Gold Creek B (kPa)</td>
<td>5865</td>
<td>8275</td>
<td>10</td>
<td>242</td>
<td>8</td>
</tr>
<tr>
<td>Berland River Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Berland River (kPa)</td>
<td>5865</td>
<td>8275</td>
<td>10</td>
<td>242</td>
<td>8</td>
</tr>
<tr>
<td>Beaver Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Beaver Creek (kPa)</td>
<td>3800</td>
<td>6040</td>
<td>10</td>
<td>225</td>
<td>8</td>
</tr>
<tr>
<td>Wolf Lake Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Wolf Lake (kPa)</td>
<td>4950</td>
<td>6450</td>
<td>10</td>
<td>151</td>
<td>8</td>
</tr>
</tbody>
</table>
## Example of Decision Variable (Table 2 of 3)

<table>
<thead>
<tr>
<th>Description</th>
<th>Min</th>
<th>Max</th>
<th>Resolution</th>
<th>N</th>
<th>String Length (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nordegg Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Nordegg (kPa)</td>
<td>4950</td>
<td>6450</td>
<td>10</td>
<td>151</td>
<td>8</td>
</tr>
<tr>
<td>Clearwater Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Clearwater (kPa)</td>
<td>4950</td>
<td>6450</td>
<td>10</td>
<td>151</td>
<td>8</td>
</tr>
<tr>
<td>Knight Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Knight (kPa)</td>
<td>3200</td>
<td>5050</td>
<td>10</td>
<td>186</td>
<td>8</td>
</tr>
<tr>
<td>Schrader Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Schrader Creek (kPa)</td>
<td>3450</td>
<td>4640</td>
<td>10</td>
<td>120</td>
<td>7</td>
</tr>
<tr>
<td>Saddle Hills Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Saddle Hills (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>Clarkson Valley Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Clarkson Valley (kPa)</td>
<td>5045</td>
<td>6895</td>
<td>10</td>
<td>186</td>
<td>8</td>
</tr>
<tr>
<td>Fox Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Fox Creek (kPa)</td>
<td>4685</td>
<td>6165</td>
<td>10</td>
<td>149</td>
<td>8</td>
</tr>
<tr>
<td>Lodgepole Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Lodgepole (kPa)</td>
<td>4685</td>
<td>6175</td>
<td>10</td>
<td>150</td>
<td>8</td>
</tr>
<tr>
<td>Vetchland Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Vetchland (kPa)</td>
<td>4950</td>
<td>6450</td>
<td>10</td>
<td>151</td>
<td>8</td>
</tr>
<tr>
<td>Dryden Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>SPC node at Dryden Creek (kPa)</td>
<td>3200</td>
<td>4430</td>
<td>10</td>
<td>124</td>
<td>7</td>
</tr>
</tbody>
</table>
### Example of Decision Variable (Table 3 of 3)

<table>
<thead>
<tr>
<th>Description</th>
<th>Min</th>
<th>Max</th>
<th>Resolution</th>
<th>N</th>
<th>String Length (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latornell Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Latornell (kPa)</td>
<td>5865</td>
<td>8275</td>
<td>10</td>
<td>242</td>
<td>8</td>
</tr>
<tr>
<td>Swartz Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Swartz Creek (kPa)</td>
<td>6290</td>
<td>8690</td>
<td>10</td>
<td>241</td>
<td>8</td>
</tr>
<tr>
<td>Pipestone Creek Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Pipestone Creek (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>Hidden Lake Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Hidden Lake (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>Meikle River B Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Meikle River B (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>Alces River B Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>DPC node at Alecs River B (kPa)</td>
<td>5000</td>
<td>8450</td>
<td>10</td>
<td>346</td>
<td>9</td>
</tr>
<tr>
<td>Vetchland Suction BV Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Edson CV Bypass BV Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Meikle River Discharge BV Status</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**Total string length:** 260

**Total possible cases:** $1.85267 	imes 10^{78}$
Step #1: Initial Population of the entire Design Space

- Use of **DOE** to populate the entire design space randomly or in an organized, analytical fashion.

- Population size is a very small fraction of the total number of cases: e.g. Upstream James River has $1.85E+78$ cases is solved with only **1000** population.

**Rules of Thumb:**
- No. of population $\geq 50$ times No. of decision variables.
This is the most challenging and diverse part of GA. Vast Literature on Constraint handling in the fitness function; for example: *Exterior Penalty Function*:

$$f(x) = \text{Obj}(x) + r \sum_{i=1}^{K} |<\phi_i>| + r \sum_{j=1}^{L} |\psi_j|$$

- $\phi_i(x) \geq 0 \quad i = 1, K$
- $\psi_j = 0 \quad j = 1, L$
- $<\alpha > = \alpha \quad \text{if} \quad \alpha \leq 0$
- $<\alpha > = 0 \quad \text{if} \quad \alpha > 0$
Other Constraint Handling Strategies:

1. Fiacco and McCormick (Exterior and Interior):

\[
f(x) = Obj(x) + r^2 \sum_{i=1}^{K} \frac{1}{\phi_i^2} + \frac{1}{r} \sum_{i=1}^{K} <\phi_i>^2 + \frac{1}{r} \sum_{j=1}^{L} \psi_j^2
\]

2. Powells (Exterior and Distorted Interior):

\[
f(x) = Obj(x) + r \sum_{i=1}^{K} <\phi_i - \sigma_i)^2 + r \sum_{j=1}^{L} (\psi_j - \tau_j)^2
\]


\[
f(x) = Obj(x) + r \sum_{i=1}^{K} [(<\phi_i - \sigma_i)^2 - \sigma_i^2] + r \sum_{j=1}^{L} [(<\psi_j - \tau_j)^2 - \tau_j^2]
\]
Fundamentals of Genetic Algorithms

Step #4: Parent Selection

Roulette Wheel Selection (among others)
Step #5: Crossover

| Chromosome 1 | 11011 | 00100110110 |
| Chromosome 2 | 11011 | 11000011110 |
| Offspring 1  | 11011 | 11000011110 |
| Offspring 2  | 11011 | 00100110110 |
Fundamentals of Genetic Algorithms

Step #5: **Crossover**

- **Single point crossover**
  - Parent A: 11001
  - Parent B: 01111
  - Offspring: 11001111
  
  \[11001011 + 1101111 = 11001111\]

- **Two point crossover**
  - Parent A: 11001
  - Parent B: 01111
  - Offspring: 10100111

- **Uniform crossover**
  - Parent A: 11001
  - Parent B: 01111
  - Offspring: 10100000

- **Arithmetic crossover**
  - Parent A: 11001
  - Parent B: 01111
  - Offspring: 10100000
Fundamentals of Genetic Algorithms

Step #6: Mutation

<table>
<thead>
<tr>
<th>Original offspring 1</th>
<th>11011111000011110</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original offspring 2</td>
<td>1101100100110110</td>
</tr>
<tr>
<td>Mutated offspring 1</td>
<td>1100111000011110</td>
</tr>
<tr>
<td>Mutated offspring 2</td>
<td>1101101100110110</td>
</tr>
</tbody>
</table>

• Mutation of a portion of offsprings is important to maintain the search space open.
Fundamentals of Genetic Algorithms

Step #7: Elitism

• A portion of best parents are copied directly into the new generation.

• Elitism can very rapidly increase performance of GA, because it prevents losing the best found solution.
Step #8: Copying

- Random copying of parents regardless of fitness into the new generation is important to maintain the search space open.
Fundamentals of Genetic Algorithms

- **Encoding**
  - Initial Population of the entire Design Space
  - Current Generation
  - Fitness Function
  - Ranking
  - Parent Selection

- **GA Operators**
  - Elitism
  - Crossover
  - Copying
  - Mutation
  - Non-Dominated Set Update (Pareto Front)

- **Converge**
  - Yes
  - No

Fundamentals of Genetic Algorithms

Non-Dominated Set (Pareto Front)

Dominated Design Case

Non-Dominated Design Case

- Linepack
Simulation Methodology

Genetic Algorithm

- Random number generated within bounds for each design variable.
- $x_1, x_2, \ldots, x_n$
- Input file
- Calculate Fitness Function
- Processed Output file
- Pareto Front
- Crossover, Mutation, Elitism, Copying.

Process Model

- Application, external executable, Code of the process
- To get Obj & violation
- Constraints.exe
- Output file
- Process output file to handle constraints and penalties.

Constraints.exe
MOGA Parameters

Parents

Children

- Elitism: 5%
- Copying: 2%
- Mutation: 5%
- Classical Cross-Over: 38%
- Directional Cross-Over: 50%
Three Examples

of Single and Multi-Objective Optimizations
Alberta System Optimization – Study Areas

North Sub-system
30 control devices
(~55 decision variables)
Many local minima

West Path Sub-system
10 control devices
(~20 decision variables)
Few local minima

<table>
<thead>
<tr>
<th></th>
<th>TransCanada</th>
<th>Alberta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Units (HP)</td>
<td>280 (5 million)</td>
<td>107 (1 million)</td>
</tr>
<tr>
<td>Pipe (Flow)</td>
<td>25,000 mi (11.5 Bcfd)</td>
<td>15,000 mi (11.5 Bcfd)</td>
</tr>
<tr>
<td>Rec/Del Points</td>
<td>1350</td>
<td>1300</td>
</tr>
</tbody>
</table>
Example 1:
Example 1:
Decision Variables

– 1 flow variable
– 2 control valve node pressures
– 8 compressor station discharge pressures
– 8 compressor station statuses (on/off)
– 1 block valve status (open/closed)

• Total of 20 decision variables
### Constraint Handling

- **Constraint penalty parameters:**

<table>
<thead>
<tr>
<th>Objective</th>
<th>Penalty Parameter Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>pressure*</td>
</tr>
<tr>
<td>Fuel Consumption (1000 m$^3$/d)</td>
<td>10</td>
</tr>
<tr>
<td>Linepack (1000 m$^3$)</td>
<td>-1,000</td>
</tr>
<tr>
<td>Throughput (1000 m$^3$/d)</td>
<td>-1,250</td>
</tr>
</tbody>
</table>

*: Units are units of objective per one percent of violation

**: Units are units of objective per one reverse flow situation

***: Units are units of objective per one percent of violation of equation shown:
Single Objective
Single Objective

- Single Objective

- Status OFF/CLOSED

- Status ON/OPEN

- Not decision variable
Two Objectives

MOGA Parameters are set to 'modified' values.
Npop = 400
Ngen = 100
A Selected Pareto Case

- **Q = 32,800**
- **Q = 65,600**
- **P = 5000**
- **P = 6185**
- **P = 7000**
- **P = 5895**
- **P = 8690**
- **P = 6135**
- **P = 6260**
- **P = 6185**
- **P = 6070**

- = Status OFF/CLOSED
- = Status ON/OPEN
- = Not decision variable
Max Linepack & Max Throughput

MOGA Parameters are set to 'modified' values.
LP constraint disabled.
Three Objectives

- Linepack (1000 m$^3$)
- Throughput (1000 m$^3$/day)
- Fuel (1000 m$^3$/day)
Example 2:

Notion of Dynamic Penalty Parameters
Example 2:
Example 2:
Decision Variables

• Decision Variables:
  - 1 flow variable
  - 6 control valve node pressures
  - 22 compressor station control pressures (suction or discharge);
  - 22 compressor station statuses (on/off)
  - 3 block valve status (open/closed)

• Total of 54 decision variables
• Searching through $10^{78}$ possible design cases
• Change the penalty scheme to incorporate a dynamic factor (DF) that can be changed with generations:

\[
fitness = \text{objective} + DF \cdot \sum R_i \cdot \text{Penalty}_i
\]

where:

\[
DF = f(\text{Generation} \#)
\]
throughput (10^3 m^3/day)

fuel (10^3 m^3/day)

- Static Penalty
- Single Objective
- Dynamic - Decreasing 1
- Dynamic - Decreasing 2
- Dynamic - Decreasing 3
- Dynamic - Decreasing 4
Computational Effort

- Computation time becomes a limiting factor as level of complexity of the system increases.
- Current processor: 2.8 GHz, 1 GB of 1066 MHz RAM, PC

<table>
<thead>
<tr>
<th>System 1</th>
<th>Compressor model</th>
<th>Population size</th>
<th>Number of generations</th>
<th>Cases run simultaneously</th>
<th>Total time required</th>
</tr>
</thead>
<tbody>
<tr>
<td>System 1</td>
<td>Block Power</td>
<td>200</td>
<td>50</td>
<td>1</td>
<td>3:10:00</td>
</tr>
<tr>
<td>System 1</td>
<td>Block Power</td>
<td>400</td>
<td>100</td>
<td>1</td>
<td>9:20:00</td>
</tr>
<tr>
<td>System 1</td>
<td>DSM</td>
<td>400</td>
<td>100</td>
<td>1</td>
<td>9:44:00</td>
</tr>
<tr>
<td>System 1</td>
<td>DSM</td>
<td>400</td>
<td>100</td>
<td>4</td>
<td>7:08:00</td>
</tr>
<tr>
<td>System 2</td>
<td>Block Power</td>
<td>500</td>
<td>100</td>
<td>1</td>
<td>23:23:00</td>
</tr>
<tr>
<td>System 2</td>
<td>DSM</td>
<td>500</td>
<td>100</td>
<td>1</td>
<td>29:58:00</td>
</tr>
<tr>
<td>System 2</td>
<td>DSM</td>
<td>500</td>
<td>100</td>
<td>4</td>
<td>17:47:00</td>
</tr>
<tr>
<td>System 2</td>
<td>DSM</td>
<td>2000</td>
<td>100</td>
<td>4</td>
<td>91:05:00</td>
</tr>
<tr>
<td>System 2</td>
<td>DSM</td>
<td>500</td>
<td>200</td>
<td>4</td>
<td>37:07:00</td>
</tr>
</tbody>
</table>
Results Validation

North Sub-system

- MOGA Single Objective
- MANUAL Single Objective
- Multi Objective

Fuel Flow

Results Validation
Third Example

- **West Path Sub-System:**
  - searching $10^{24}$ possible cases
  - single and two-objective
  - static penalty produced satisfactory results
Computational Effort

- Computation time becomes a limiting factor as level of complexity of the system increases.
- Current processor: 2.8 GHz, 1 GB of 1066 MHz RAM, PC

<table>
<thead>
<tr>
<th>Sub-system</th>
<th>Single Objective</th>
<th>Two Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORTH</td>
<td>GA, 500x200 static penalties</td>
<td>GA, 1000x200 dynamic penalties</td>
</tr>
<tr>
<td>Sub-system</td>
<td>~2 days</td>
<td>~4 days</td>
</tr>
<tr>
<td>WPATH</td>
<td>GA, 500x100 static penalties</td>
<td>GA, 1000x200 static penalties</td>
</tr>
<tr>
<td>Sub-system</td>
<td>~5 hours</td>
<td>~1 day</td>
</tr>
</tbody>
</table>
Reducing Computation Time

• **Option A**: reduce number of design cases evaluated

• **Option B**: simplify model to reduce computational effort per case

• **Option C**: more computer power

• Combination of above
Surrogate Methods

- **A combination of ‘A’ and ‘B’:**
  - Hydraulic model:
    - High-fidelity model,
    - high computational effort.
  - Surrogate model:
    - Low-fidelity model,
    - lower detail,
    - “training” required,
    - low computational effort.
Surrogate Methods

- Test three low-fidelity (surrogate) models:
  - Kriging
  - Neural Networks
  - Quadratic Response Surface

- Training data from a short GA run (e.g. best population after 10 generations)
After 10 Gen. - Unpenalized Design Cases

flow ($10^3$ m$^3$/day) vs. fuel ($10^3$ m$^3$/day)
Surrogate Methods

- Each model was “trained”.
- 50 generations using surrogate.
- Best results fed back into additional 10 generations using hydraulic simulator.
Flow vs. Fuel - Results from using Quadratic RSM Surrogate
Flow vs. Fuel - Results from using Kriging Surrogate

- 200 gen. Hi-Fi
- 20 gen. Hi-Fi
- Final - 10-50-10

fuel ($10^3 \text{m}^3$/day) vs. flow ($10^3 \text{m}^3$/day)
Flow vs. Fuel - Results from using Neural Network Surrogate

- 200 gen. Hi-Fi
- 20 gen. Hi-Fi
- Final - 10-50-10
Computational Effort

- **Benefit may not be significant:**

<table>
<thead>
<tr>
<th></th>
<th>10 Hi-Fi</th>
<th>50 Lo-Fi</th>
<th>10 Hi-Fi</th>
<th>Total (hrs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quad. RSM</td>
<td>1.57</td>
<td>0.02</td>
<td>1.57</td>
<td>3.15</td>
</tr>
<tr>
<td>Neural Net</td>
<td>1.57</td>
<td>2.47</td>
<td>2.43</td>
<td>6.47</td>
</tr>
<tr>
<td>Kriging</td>
<td>1.57</td>
<td>1.30</td>
<td>1.45</td>
<td>4.32</td>
</tr>
<tr>
<td>20 Gen Hi-Fi only</td>
<td></td>
<td></td>
<td></td>
<td>3.05</td>
</tr>
<tr>
<td>200 Gen Hi-Fi only</td>
<td></td>
<td></td>
<td></td>
<td>35.00</td>
</tr>
</tbody>
</table>
Conclusions

• Application of GA is robust.
• Careful consideration to GA “tuning” parameters and proper constraint handling
• Application of surrogate methods can improve computational time.
Acknowledgment

Jeff Stoffregen
Hossein Golshan
Krystyna Jungowski
Gene Poissant
Daniel Sennhauser
Landen Stein
Thomas Robinson
&
TransCanada: for granting permission to present results