EC Applications

- Systems Identification
- Economic Dispatch
- Time Series Artificial Neural Networks
- Circle Packing
- Hidden Markov Models

EC Applications

- Representation
- Domain Knowledge
- Fitness Function
- Constrained Optimization

Systems Identification

- Black-Box Identification
  - Identify model form
  - Identify model’s parameters
- Gray-Box Identification
  - Model form known
  - Identify model’s parameters
- Modelling for control

Systems Identification

- u – input, y – output,
  \[ y(t) + a_n y(t-1) + \ldots + a_2 y(t-n) = b_n y(t-n) + b_1 y(t-n) + \ldots + b_1 y(t-n) + \varepsilon(t) \]
- Chromosome structure
  \[ \theta = [a_1, a_2, \ldots, a_n, b_1, \ldots, b_n]^T \]
- Fitness function
  \[ \text{error} = \sqrt{\frac{1}{N-1} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2} \]

Systems Identification

- Typical control theory modelling tool: transfer functions
  \[ A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_n q^{-n} \]
  \[ q^{-1} y(t) = y(t-1) \]
  \[ B(q) = b_1 q^{-1} + \ldots + b_n q^{-n} \]
  \[ G(q, \theta) = \frac{B(q)}{A(q)} \]
  \[ H(q, \theta) = \frac{1}{A(q)} \]
Systems Identification

• $R_1 = 25.08 \, \text{k\Omega}$, $R_2 = 25.1 \, \text{k\Omega}$, $R_3 = 25.09 \, \text{k\Omega}$, $R_4 = 25.99 \, \text{k\Omega}$, $C_1 = 0.517 \, \mu\text{F}$, $C_2 = \mu\text{F}$, $C_3 = 0.485$, $C_4 = 0.55 \, \mu\text{F}$

• Real model

\[ H(q) = \frac{29.75 \times 10^{-3}q^4 + 223.0 \times 10^{-5}q^2 + 154.9 \times 10^{-3}q + 9.418 \times 10^{-5}}{q^4 - 2.657q^2 + 2.565q^2 - 1.011q + 0.147} \]

Systems Identification

What did we get?

Systems Behaviors

Simulation

Identification

Economic Dispatch

• An electrical network
• $n$ generators
• $m$ loads
• How much does each generator should produce to minimize cost $= C$?

\[ \begin{align*}
\min_{P} & \quad C = \sum_{i \in \Omega} c_i(P_i) \\
\text{subject to} & \quad \left( \sum_{i \in \Omega} P_i \right) - P_D - P_L = 0 \\
& \quad P_i^\text{min} \leq P_i \leq P_i^\text{max} \quad \forall i \in (\Omega) \\
& \quad c_i(P_i) = \alpha_{i,0} + \alpha_{i,1}P_i + \alpha_{i,2}P_i^2 + \cdots + \alpha_{i,n}P_i^n
\end{align*} \]

Economic Dispatch

• Chromosome design $\theta = [P_1, \ldots, P_n]$

• Fitness function $f = \sum P_i$

• How to stay in the constrained region?
Economic Dispatch

- Punish
- Repair
- Vector Combination
- DE
- PSO

Predictions

It is difficult to make predictions, especially about the future.

*Neils Bohr*

"The population is constant in size and will remain so right up to the end of mankind." J. Encyclopedia, 1756.

"1930 will be a splendid employment year," U.S. Department of Labor, New Year's Forecast in 1929, just before the market crash on October 29.

"Computers are multiplying at a rapid rate. By the turn of the century there will be 220,000 in the U.S." Wall Street Journal, 1966.

Time Series Forecasting

- What is the next value?
- The next k values?

\[
\hat{y}_t = \sum_{k=1}^{w} a_k y_{t-k} + \sum_{k=1}^{w} b_k e_{t-k} + e_t
\]

\[
e_t = \hat{y}_t - y_t
\]

Evolving ANN Models

- Input-output similar to AR
- Architecture of an Artificial Neural Network:
  - 3-Layer perceptron.
- ANN design using EC:
  - Architecture Design
  - Weight Design

Evolving ANNs
Evolving ANNs

- Chromosome of the outer evolutionary process

<table>
<thead>
<tr>
<th>Vars</th>
<th>NH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 1</td>
<td>1 0 1</td>
</tr>
</tbody>
</table>

- 0 indicates the variable is not taken into account
- 1 indicates the variable is taken into account
- NH - neurons in the hidden layer (binary-coded integer)

Evolving ANNs

- Chromosome of the inner evolutionary process.

- Vector of real numbers (weights of the synaptic connections)

Circle packing Problem - CPP

Given $n$ circles $c_i, i \in \{1, \ldots, n\}$, find the diameter if the smallest containing circle $c_0$, such that all circles fit in the container circle without overlap.

CPP

- Container $c_0$, radius $r_0$
- $c_i$ centered at $(x_i, y_i)$ and radius $r_i$, $i \in \{1, \ldots, n\}$
- Determine the smallest radius $r_0$, coordinates $(x, y)$ such that

\[
\sqrt{x^2 + y^2} \leq r_0 - r_i, \quad i \in N
\]
\[
\sqrt{(x_i - x)^2 + (y_i - y)^2} \geq r_i + r_j, \quad i \neq j \in N
\]

CPP

- Fixing the position of the $n$ circles, we have a configuration
- $X=(x_1, y_1, x_2, y_2, \ldots, x_n, y_n)$
- If $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} < r_i + r_j$, $c_i$ and $c_j$ overlap
Chromosome = configuration

\((x_1, y_1, x_2, y_2, ..., x_n, y_n)\)

Random generation = overlaps

Individual with no overlaps + mutation = individual with overlaps

Parents with no overlaps + crossover = offspring with overlaps

Mutation produces overlaps

Xover produces overlaps
Search vs. Feasible Spaces

• Xover produces overlaps

Search vs. Feasible Spaces

• Constraint violations
• Punishment
• Repair

Search vs. Feasible Spaces

Search vs. Feasible Spaces

Search vs. Feasible Spaces

Search vs. Feasible Spaces

Search vs. Feasible Spaces

• Working in search space -> wasted work
• Solution: remain in feasible space
• Means: repair
Anomaly Detection (AD)

- Anomaly Detection is performed by detecting:
  - Behavior of the system
  - Resources utilization
  - Changes in the pattern
- A statistical model contains metrics derived from system operation ("normal" behavior)
- Any behavior that varies from this model is considered an anomaly

HMMs...

- An HMM is formed by a finite number of states connected by transitions
- HMMs can generate an observation sequence depending on its transitions, and probabilities

HMMs...

- An HMM $\lambda$ is defined by the triplet:
  $\lambda = (A, B, \pi)$
  - $\pi$ = Initial State Probability.
  - $A$ = State Transition Probability Matrix.
  - $B$ = Emission Probability Matrix.
- A Markov Model is hidden because we do not know which state led to each observation.

Evolutionary Programming (EP)

- Genetic Programming is a global search technique, that can be used to optimize the HMM parameters.

<table>
<thead>
<tr>
<th>Size</th>
<th>Transitions</th>
<th>Parameters</th>
<th>$\Pi$</th>
</tr>
</thead>
</table>

EP...

- The fitness value is calculated by the objective function:
  $P(O|\lambda) = \sum_{i=1}^{N} \alpha_i(i)$
- Forward probability: observation sequence had been generated by the HMM
Evolving HMMs

Results

• EP evolved HMMs based on the observation sequence given by the network bandwidth used in our University.

Results...

• We took the HMM with the highest probability
• We combined the HMM and a Time Window data to detect anomalies in our tests
• This two elements determine the probability of a given sequence had been generated by the HMM

Results...

• Our test consists of the following:
  • Given a time window data, the HMM analyzes it and determines the probability of the observation sequence.
  • The time window data moves to the next item in the observation sequence, which is also verified by the HMM.
  • The process continues until the whole observation sequence is traversed and tested.

Results...

• All probabilities generated by the sliding time windows and the HMM, were compared against a probability threshold
• We analyzed these probabilities and determine if an anomaly exists or not

Results...

<table>
<thead>
<tr>
<th>Window Size</th>
<th>Hits</th>
<th>False Positives</th>
<th>False Negatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>119</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>93</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>89</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>84</td>
<td>5</td>
<td>11</td>
</tr>
</tbody>
</table>

Probability Plot for Window Size 4
## Conclusions

- Representation – not always straightforward
- Fitness Function – not always cheap
- Constrained Optimization
  - punishment vs. repair
  - wasting CPU cycles – getting nowhere
- Automated production of models