CS 4752
Introduction to Computational Intelligence
Lecture 7
September 22, 2011

Outline

• Deterministic Selection
• Population Size
• Running Computer Experiments
  – Modifying existing systems
• Reporting Results
  – Multiple Runs
  – Mean and standard error (error bars)
  – Statistical test?
• Plotting Results
Deterministic Uniform

• In Evolutionary Programming, each parent is selected exactly once to produce once offspring.
• In stochastic uniform (random) selection, the best individual may never get selected while the worst may get selected multiple times.
• Deterministic uniform can avoid such bias.

Deterministic Replacement

• In steady-state model, the replacement of current population individual can be decided deterministically:
  – Age-based: replace the oldest
  – Fitness-based: replace the worst
• Advantage/disadvantage compared to random replacement?
Truncation - Elitism

- The best \(b\) individuals in the population are selected.
- **Advantages**
  - prevents the best found candidate solution accidentally 'gets lost'
  - preserves the currently best found solution as fix points to create offspring in their vicinity
- **Disadvantage**
  - May cause premature convergence

Truncation Implementation

- **Evolution Strategies (\(\mu,\lambda\))-ES**
  - \(\mu\) is the size of parent population while \(\lambda\) (\(\lambda >> \mu\)) is the size of the offspring population.
  - Randomly select parents to generate \(\lambda\) offspring
  - The best \(\mu\) individuals among \(\lambda\) offspring form the new population (non-overlapping population).

\[
P(t) \quad \mu \text{ parents} \quad \lambda \text{ offspring} \quad P(t+1) \quad \mu \text{ parents}
\]
Truncation Implement II

- Evolution Strategies (µ+λ)-ES
  - µ is the size of parent population while λ is the size of the offspring population.
  - Randomly select parents to generate λ offspring.
  - The best µ individuals among λ+µ form the new population (overlapping population).

```
<table>
<thead>
<tr>
<th>P(t)</th>
<th>µ parents</th>
</tr>
</thead>
<tbody>
<tr>
<td>reproduction</td>
<td></td>
</tr>
<tr>
<td>µ parents</td>
<td>λ offspring</td>
</tr>
<tr>
<td>P(t+1)</td>
<td>µ parents</td>
</tr>
</tbody>
</table>
```

Selection – Only Model

- Under **fitness-based selection**, population fitness improve over the generations.
- However, the best individual fitness at the last generation can never be better than the best individual fitness at the initial generation.
- Moreover, under **stochastic selection**, the best individual may get lost and the population converged to an individual with worse fitness than that of the initial best individual.
Population Size

• One way to increase the probability of converging to high fitness individuals is by increasing the population size:
  – Better coverage of the search space (diversity) which helps high fitness individuals to be included in the initial population
  – A larger past memory, so that good individuals are not lost so quickly

Running Computer Experiments

• Multiple Runs:
  – I use ancient technologies to conduct batch runs.
  – sh script, awk programs
  – See run.sh and process.awk on the class website.
  – You can write your own script to make batch runs.
Statistical Analysis

• We only use simple toy problems in the class, hence I don’t ask you to conduct statistical test (e.g. Mann-Whitney-Wilcoxon test).
• In most real world problems, you will need to conduct statistical tests to support your analysis.
• Nevertheless, you need to report the mean and standard error of your experimental results.

Excel Example

• Standard error = standard deviation / \sqrt{n}

![Average Population Fitness](image)