Data-Driven Evolutionary Optimization of Complex Systems: Big vs Small Data

Yaochu Jin

Head, Nature Inspired Computing and Engineering (NICE)
Department of Computer Science, University of Surrey, United Kingdom

yaochu.jin@surrey.ac.uk
• Complexity in evolutionary optimization of real-world problems

• Data driven evolutionary optimization
  – Offline (Big) data driven evolutionary optimization
  – Online small data driven evolutionary optimization

• Concluding remarks
Strengths and Weaknesses of EAs for Optimization

- No need for analytical objective functions and no requirement for derivative information
- Less vulnerable to local optimums
- Less vulnerable to uncertainty (relative quality is more important)
- Well suited for multi-objective optimization
- No theoretical guarantee for global optimum
- Population based search -- computationally intensive
Complexities in Real-World Optimization

• Complexity in problem formulation and solution representation

• Complexity in scale
  ➢ Large number of decision variables
  ➢ Large number of objectives

• Complexity in handling uncertainty

• Complexity in quality evaluation
Complexity in Problem Formulation and Solution Representation
The formulation of the objective function is an iterative process.

- Representation is critical: multiple sub-systems, optimization-control coupling.
- Different objectives/constraints/decision variables may have to be considered/weighted differently at different stages.
- Different resources are available at different stages.
Complexity in Shape Representation

Direct representation

Expert Knowledge

Free Form Deformation (FFD)

Bezier Curves, B-Splines and NURBS
Shape Representations in Micro Heat Exchanger

A Spline representation

A frequency-amplitude representation
Micro Heat Exchanger Optimization - Results

- Maximize the heat transfer rate (thermodynamic)
- Minimize the sum of pressure drop with a penalty (aerodynamic)
Complexity in Scale:
Large Decision and Objective Space
Swarm Intelligence for Large-Scale Optimization

- Large-scale evolutionary optimization
  - Divide and conquer by random grouping
  - Detection of correlation between decision variables
  - A competitive swarm optimizer (CSO)
  - A social-learning based particle swarm optimizer (SL-PSO)

Competitive Swarm Optimization (CSO)

- Randomly pick two solutions
- Compare their fitness. The winner is directly passed to the next generation while the loser will be updated as follows:

\[
V_{l,k}(t + 1) = R_1(k, t)V_{l,k}(t) \\
+ R_2(k, t)(X_{w,k}(t) - X_{l,k}(t)) \\
+ \varphi R_3(k, t)(\bar{X}_k(t) - X_{l,k}(t))
\]

- Neither global nor personal best is used
A Social Learning PSO (SL-PSO)

SL-PSO Main Loop

Fitness Evaluation → Swarm Sorting → Behaviour Learning

Swarm $P(t)$ $\rightarrow$ $t = t + 1$ $\rightarrow$ Swarm $P(t+1)$

Swarm before sorting

\[ 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad \ldots \quad m \]

Sort according to fitness values

\[ \text{Worst} \quad \text{Swarm after sorting} \quad \text{Best} \]

\[ 1 \quad 2 \quad \ldots \quad i \quad k \quad m \]

Demonstrators

\[
X_{i,j}(t+1) = \begin{cases} 
X_{i,j}(t) + \Delta X_{i,j}(t + 1), & \text{if } p_i(t) \leq P_i^L \\
X_{i,j}(t), & \text{otherwise}
\end{cases}
\]

\[
0 \leq p_i(t) \leq P_i^L \leq 1
\]

\[
\Delta X_{i,j}(t + 1) = r_1(t) \cdot \Delta X_{i,j}(t) + r_2(t) \cdot I_{i,j}(t) + r_3(t) \cdot \epsilon \cdot C_{i,j}(t),
\]

\[
\begin{cases}
I_{i,j}(t) = X_{k,j}(t) - X_{i,j}(t), \\
C_{i,j}(t) = \bar{X}_j(t) - X_{i,j}(t).
\end{cases}
\]
Large Number of Objectives – Many-Objective Optimization

- Computational challenges
  - Calculation of performance some indicators becomes intractable

- Performance degradation
  - Loss of selection pressure in Pareto-based approaches

- Solution assessment becomes tricky
  - The performance become very sensitive and also easily biased
  - Solution sets are no loner comparable
  - Diversity becomes trickier to measure

- Can we still be able to find a “representative” subset of the Pareto front?

EAs for solving MaOPs may largely be divided into the following categories:

- **Preference based**, including decomposition approaches

- **Convergence acceleration**, mainly by modifying the dominance relationship or by **including additional criteria**

- Performance indicator based
Use of “Knee-Points” to Accelerate Convergence

Specification of Preferences

Angle-penalized distance (APD):

\[ d^j = (1 + P(\theta^j)) \cdot \| \bar{f}^j \|, \]

\[ P(\theta^j) = k \cdot \left( \frac{t}{t_{\text{max}}} \right)^\alpha \cdot \frac{\theta^j}{\gamma_v}, \]

Efficient Non-Dominated Sorting

- Non-dominated sorting becomes extremely time-consuming in case of
  - A large population size
  - A large number of objectives

- Computationally efficient non-dominated sorting
  - ENS: An efficient non-dominated sorting algorithm for 2 or 3 objectives with a large population size
  - A-ENS: An approximate non-dominated sorting for many objectives
  - T-ENS: An accurate tree-based non-dominated sorting for large-scale many-objective optimization

X. Zhang, Y. Tian, Y. Jin. Approximate non-dominated sorting for evolutionary many-objective optimization. *Information Sciences*, 2016 (accepted)
X. Zhang, Y. Tian, R. Cheng, and Y. Jin. A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. 2016 (Submitted)

Code for the ENS variants available!
Complexity in Quality Evaluation
Complexity in Quality Evaluation

- An analytic fitness function is not available
  - Very time-consuming numerical simulations
  - Expensive experiments
  - History production data only
Data Driven Evolutionary Optimization
Data Driven Optimization – Offline and Online

Data-driven evolutionary optimization

Offline data-driven optimization

Offline (Big) Data Driven Evolutionary Optimization
Scottish Trauma system design

- A specialist hospital responsible for providing a regional or national service for patients with major trauma.

- A trauma unit is a hospital which manages less severely injured patients.

- A local emergency hospital is a hospital which only deals with minor injuries.

Objective 1: Minimize total travel time

Objective 2: Minimize the exceptions number (the cases "triaged-to-MTC" diverted to a TU)

Constraint 1: Number of helicopter transfers

Constraint 2: MTC case volume

Constraint 3: TU proximity to avoid two close TUs

Main research question is: how to reduce computational time given the large amount of data (the amount of data could be much larger)?

- The EA is able to find the Pareto optimal solutions
- The computation time can be reduced as much as possible
• 40,000 incident records (location, injury, patient)
• 18 trauma centers in Scotland

• Fitness evaluations is highly time-consuming when the number of records is huge
Elitist Non-dominated Sorting GA (NSGA-II)

1. Merge
2. Non-dominated sorting
3. Crowding distance sorting
4. Crowded Non-dominated sorting

Terminate

Evaluate (Offspring)

Mutate

Recombine

Select

Evaluate

Initialize

Non-dominated sorting

Non-dominated sorting
Solution

- Group the data into a number of clusters and use the cluster centers to evaluate the objective and constraint functions.
- How to make sure the accuracy is good enough?

The maximum error should not change the individuals to be selected.

Estimated error for a given number of clusters:

\[ ER = \frac{1}{\beta_1 + \beta_2 K} \]

First and last front of individuals to be selected.
• Adaptation of K
Data Driven Evolutionary Trauma System Design

<table>
<thead>
<tr>
<th></th>
<th>IGD</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA-NSGA-II with CS</td>
<td>$2.47e-02 \pm 1.96e-02$</td>
<td>$6.54e+03 \pm 3.57e+03$</td>
</tr>
<tr>
<td>SA-NSGA-II without CS</td>
<td>$2.51e-02 \pm 2.79e-02$</td>
<td>$10.31e+03 \pm 5.00e+03$</td>
</tr>
<tr>
<td>NSGA-II</td>
<td>$0.00e+00 \pm 0.00e+00$</td>
<td>$8.14e+04 \pm 5.36e+02$</td>
</tr>
</tbody>
</table>
Discussions

- Online big data driven evolutionary optimization, e.g., stream data
- Efficient learning of big data
- Noisy and / or heterogeneous data
Online (Small) Data Driven Evolutionary Optimization
In many cases, collecting data is very expensive

- Highly time-consuming numerical simulation
- Expensive physical experiments
- Real-industrial processes only
Surrogate-Assisted Evolutionary Optimization

- Use a meta-model/surrogate to replace the expensive fitness evaluations, e.g., CFD simulations
  - Choose a surrogate
  - Collect data
  - Train the surrogate
  - Replace the CFD

Online data-driven optimization

CFD simulations / Experiments
Model Management

- Use surrogates only: Risk of converging to a false optimum

- Surrogate management / evolution control
  - population-based
  - generation-based
  - individual-based
  - local search
  - Combination of the above

Model Management – Main Questions

Which individuals are to be re-evaluated using the real-fitness function?

• Solutions that are of potentially good performance
• Solutions whose estimated fitness have large amount of uncertainty
  ➢ Less explored
  ➢ Effective for model improvement

• How to measure uncertainty?

• How to measure model quality? (Jin et al 2003, Huesken et al 2005)

The best strategy is more efficient than the random strategy.

In the best strategy, about half of the individual should be controlled to guarantee correct convergence.

Model Management – Promising and Uncertain Ones

Given a stochastic model (Gaussian process),

• Mean fitness value:

\[ f = \mu(x); \]

• Lower confidence bound (LCB)

\[ f = \mu(x) - \alpha \sigma(x) \quad (\alpha > 0) \]

• Expected improvement (EI)

\[
\text{EI}(x) = \begin{cases} 
(\mu(x) - f(x^+))\Phi(Z) + \sigma(x)\phi(Z) & \text{if } \sigma(x) > 0 \\
0 & \text{if } \sigma(x) = 0
\end{cases}
\]

\[ Z = \frac{\mu(x) - f(x^+)}{\sigma(x)} \]

• Probability of Improvement (PI)

\[ \Phi(\cdot) \text{ is the normal cumulative distribution function.} \]

\[ \text{PI}(x) = P(f(x) \geq f(x^+)) = \Phi \left( \frac{\mu(x) - f(x^+)}{\sigma(x)} \right) \]


Model Management in Multi-Objective Optimization

- Each objective is considered separately (similar to single objective optimization)

- Multiple objectives are converted to a scalar objective function and then use the model management criteria for single objective optimization
  - Random weights
  - Uniformly distributed weights

- Use a scalar performance indicator, e.g., hypervolume

Potential Benefit of a Global Model

A global model might help smoothen the fitness landscape


Dual Surrogates in Memetic Algorithms

START

Initialize and evaluate a parental population

Create offspring population by applying evolutionary operator on parents

Evaluate all individuals using original fitness function

Perform local refinements on all individuals

Termination condition reached?

Yes

No

Select new parental population

END

For each individual $i = 1, 2, \ldots, \text{popsize}$

Methods to generate a more robust surrogate model:
- ensemble of multiple surrogate models
- gradient-based surrogate model
- etc.

Build the first local surrogate model, $M_1$, which can provide robust prediction accuracy

Perform local refinement using $M_1$

Build other local surrogate model(s), $M_2, M_3, \ldots, M_k$ to facilitate greater diversity in the search

Perform local refinement(s) using each of $M_2, M_3, \ldots, M_k$

Replace every individual with the best local optimum found from search in $M_1, M_2, M_3, \ldots, M_k$ using Lamarckian/Baldwinian learning

Smooth

Exploiting

Dual Surrogate Single Objective MA (DS-SOMA)
Results: Multi-Objective

1. DS-MOMA
2. SS-MOMA-PR+RBF+GP
3. SS-MOMA-PR
4. SS-MOMA-Perfect

All results from 20 runs
Global and Local Surrogate Models

Two-layer (global and local) surrogate-assisted PSO

1: Construct a global surrogate model;
2: Approximate a fitness value for each individual in the swarm using the global surrogate model;
3: for each particle $i$ in the swarm do
4: Find its neighbors in the local database;
5: if there are enough samples to construct a local surrogate then
6: Construct a local surrogate model;
7: Approximate the fitness of particle $i$ using the local surrogate model;
8: $\tilde{f}(x_i) = \min\{\tilde{f}_g(x_i), \tilde{f}_l(x_i)\}$
9: else
10: $\tilde{f}(x_i) = \tilde{f}_g(x_i)$
11: end if
12: end for

Surrogate-Assisted Large-Scale Evolutionary Optimization?

- Dimension mostly limited up to 10 in GP-assisted EAs, mostly up to 30

  - Curse of dimensionality

  - Dramatic increase in computational cost for training surrogates, e.g., Gaussian processes – it can take hours to build a GP model 😞
Fitness Estimation Assisted CSO for Large-Scale Optimization

- Fitness estimation in competitive swarm optimization for dimensions up to 500
  - Fitness estimated based on positional relationships

Concluding Remarks

- Evolutionary optimization of complex systems is promising yet challenging

- Data-driven evolutionary optimization becomes increasingly important

- Surrogate-assisted evolutionary optimization will not only be essential for evolutionary optimization of complex systems but also provides a platform for integrating evolution and learning techniques
  - Which to re-evaluate (sample)
    - Active learning
  - Small data
    - Semi-supervised learning
    - Transfer learning
  - Big data
    - Deep learning