Optimization of Single Screw Extrusion

António Gaspar-Cunha
Institute of Polymers and Composites,
Department of Polymer Engineering, University of Minho
Guimarães, Portugal

Contents

• The problem to solve
• Optimization methods
• Evolutionary algorithms
• Multi-objective evolutionary algorithms
• Results
**Extrusion modelling**

**INPUT DATA:**
- Geometry
- Polymer Properties
- Operating Conditions

**RESULTS:**
- Output
- Power consumption
- Melt temperature
- Temperature homogeneity
- Degree of Mixing
- Length required for melting
- Pressure generation
- ...
Approaches to optimize the processes (e.g., set the operating conditions, design machines, etc…):

- Use empirical knowledge;
- Use computational tools on a trial and error basis;
- Solve the inverse problem;
- Perform a partial process optimization;
- Develop a global optimization procedure.
### Optimization Methods

**Solve the Inverse Problem**

#### Direct Problem:
- Geometry
- Material properties
- Operating conditions

Governing equations

#### Inverse Problem:
- Material properties
- Output
- Power consumption
- Melt temperature
- Degree of mixing

Governing equations

### Optimization Methods

**Develop a Global Optimization Procedure**

- Optimization Algorithm
- RESULTS

- USER INTERFACE

- Objective Function

- Evaluation/new (better) solutions

- Performance Machine/tool response

- Modelling Package
optimization methods

\[ f(x, y) \]

[Graph showing optimization methods]

optimization methods

[Graph showing optimization methods with a red cross]
optimization methods

OPTIMIZATION ALGORITHMS

- Random search
- Gradient methods
- Simulated annealing
- Neural networks
- Expert systems
- Sensitivity analysis
- Statistical methods
- Ant colony optimization
- Evolutionary algorithms
The role of optimisation is to find the best set of parameters that optimise an objective function, particularly by improving the performance in the direction of some optimal point or points:

\[
\text{maximise}_{x \in \Omega} \quad f(x) \quad \text{subject to} \quad g_j(x) \geq 0 \quad j = 1, \ldots, J \\
\quad h_k(x) = 0 \quad k = 1, \ldots, K
\]

where \( x \) is a vector of \( n \) and \( \Omega \subseteq \mathbb{R}^n \), \( \Omega = \{ x \in \mathbb{R}^n : l \leq x \leq u \} \).

\( f \) is the objective function of the \( n \) parameters \( x \), \( g_j \) are the \( J \) \( (J \geq 0) \) inequality constraints, and \( k \) are the \( K \) \( (K \geq 0) \) equality constraints.
Evolutionary Algorithms (EAs) are stochastic search and optimisation methods that mimic natural evolution through genetic operators like crossover and mutation.

- They work with a population of points, each one representing a possible solution in the search space.
- Each individual has a value associated to it (fitness or objective function), which is a measure of its performance on the system.
- Individuals with greater performance have a bigger opportunity for reproduction, i.e. to pass their characteristics to future.

### Evolutionary Computation – THE METAPHOR

<table>
<thead>
<tr>
<th>Natural Evolution</th>
<th>Evolutionary Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual</td>
<td>Solution</td>
</tr>
<tr>
<td>Performance</td>
<td>Quality</td>
</tr>
<tr>
<td>Environment</td>
<td>Problem</td>
</tr>
</tbody>
</table>
Evolutionary computation is an iterative technique, i.e., successively new populations are generated until a good solution is found.

The Computational Cycle

Random initialization (or semi-random) of the population of candidate solutions (Generation 0)

Evaluation of the performance of population individuals

Generation of a new population from members with more performance through genetic operations (recombination and mutation)

The Evolutionary Cycle

Population

Selection

Parents

Recombination

Mutation

Offspring

Replacement
**Multi-objective Evolutionary Algorithms**

Most real optimization problems are multi-objective

**Example:** Simultaneous minimization of the cost and maximization of the performance (comfort) when buying a car

![Diagram](image)

- Single optimum (minimizing the cost)
- Multiple optima (optimizing both objectives)
- Single optimum (maximizing the performance)

---

**Non-Dominance Concept**

maximize $f_1(x_i), f_2(x_i)$

- Better
- Indifferent
- Worst

Pareto optimal (solutions non-dominated)

Dominated
multi-objective evolutionary algorithms

Multi-Objective Optimization Problem

\[
\begin{align*}
\min_{x_i} & \quad f_i(x) & i = 1, \cdots, n \\
\text{subject to} & \quad g_j(x_i) = 0 & j = 1, \cdots, J \\
& \quad h_k(x_i) \geq 0 & k = 1, \cdots, K
\end{align*}
\]

where \( f_i \) are the \( M \) objective functions of the \( n \) parameters \( x_i \), and \( g_j \) and \( h_k \) are the \( J \) equality \((J \geq 0)\) and \( K \) inequality \((K \geq 0)\) constraints, respectively.

multi-objective evolutionary algorithms

Decision making

Pareto Optimality (a set of optimal trade-offs, all objectives have equal importance)

Decision Making (choose the best compromise based on preference information)
**Multi-Objective Optimization**

**Step 1: Find the Pareto-optimal solutions**

Multi-Objective Optimization Problem: Maximize: \( f_1, f_2, \ldots, f_n \)

- Multi-objective optimizer
- Find the Pareto-optimal solutions

**Step 2: Select a single solution**

- Multi-objective optimizer
- Preference information
- Single solution
- Multiple solutions

**Evolutionary Algorithm**

- **Start**
- Initialise Population
  - \( i = 0 \)
  - Evaluation
  - Assign Fitness \( F_i \)
  - Convergence criterion satisfied?
    - yes
      - Selection
      - Recombination
    - no
  - \( i = i + 1 \)
- **Stop**

**Population:** set of chromosomes or individuals

**Chromosome**

<table>
<thead>
<tr>
<th>L1</th>
<th>L2</th>
<th>D1</th>
<th>D3</th>
<th>e</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Initialization of population:** random definition of all individuals of the population

**Evaluation:** calculation of the values of the objectives using the modeling routine

**Fitness:** calculation of a single value identifying the performance of individual

**Reproduction:** selection of the best individuals for crossover and/or mutation

**Crossover/Mutation:** methods to obtain new individuals for the next generation \((i+1)\)
Optimization Methodology

- Numerical/modelling routines
  - Structural mechanics
  - Fluid dynamics
  ...
- Evaluation of solutions

Problem Characteristics

Optimization Algorithm

END

multi-objective evolutionary algorithms

results

i) Hopper

N = [10, 60] rpm

2.7D

Pitch: P = [30, 42] mm

Flight thickness: e = [3, 4] mm

D1 = [20, 26] mm

L1 = [100, 400] mm

L2 = [170, 400] mm

L = 936 mm

L = 936 mm

Heater bands

Tb = [150, 210] °C

Tb = [150, 210] °C

Heater bands

Heater bands

Die

Die

D3 = [26, 32] mm

D3 = [26, 32] mm

20 mm

20 mm

35 mm

35 mm

Pitch:
P = [30, 42] mm

Pitch: 
P = [30, 42] mm

Flight thickness:
e = [3, 4] mm

Flight thickness: 
e = [3, 4] mm

L = 936 mm

L = 936 mm

N = [10, 60] rpm

N = [10, 60] rpm

L = 936 mm

L = 936 mm

D = [20, 30] mm

D = [20, 30] mm

D = [20, 30] mm

D = [20, 30] mm

Output – Q (kg/hr)

Maximize 1 20

Length for melting – L (m)

Minimize 0.2 0.9

Melt temperature – T (°C)

Minimize 150 210

Power consumption – P (W)

Minimize 0 9200

WATS

Maximize 0 1300

Objectives | Aim | X_{min} | X_{max} |
---|---|---|---|
Output – Q (kg/hr) | Maximize | 1 | 20 |
Length for melting – L (m) | Minimize | 0.2 | 0.9 |
Melt temperature – T (°C) | Minimize | 150 | 210 |
Power consumption – P (W) | Minimize | 0 | 9200 |
WATS | Maximize | 0 | 1300 |
WHAT DO WE NEED FROM PREVIOUS STEPS OF THE PROJECT?

- Polymer properties
- Operating conditions
- System geometry
**WHAT DO WE NEED FROM PREVIOUS STEPS OF THE PROJECT?**

<table>
<thead>
<tr>
<th>Polymer properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Friction coefficients (very important);</td>
</tr>
<tr>
<td>• Solid density = f(P,T);</td>
</tr>
<tr>
<td>• Melt density = f(P,T) − PVT;</td>
</tr>
<tr>
<td>• Thermal conductivity (solid and melt);</td>
</tr>
<tr>
<td>• Heat capacity (solid and melt);</td>
</tr>
<tr>
<td>• Heat of fusion;</td>
</tr>
<tr>
<td>• Melting temperature;</td>
</tr>
<tr>
<td>• Viscosity = f(T, shear rate)</td>
</tr>
</tbody>
</table>

---

**Thanks!**

**Debate**