13 Large-scale Global Optimization

李长河

中国地质大学（武汉）自动化学院

lichanghe@cug.edu.cn

李长河

自动化学院 710

lichanghe@cug.edu.cn

Large-scale Global Optimization

Complex Optimization Problems

- One of the manifestations of complexity is that the increase in the number of variables causes the dimensionality disaster.
- Large-scale global optimization (LSGO) problems usually contains more than 1000 variables.

Large-scale Global Optimization

Complex Optimization Problems

- One of the manifestations of complexity is that the increase in the number of variables causes the dimensionality disaster.
- Large-scale global optimization (LSGO) problems usually contains more than 1000 variables.

Application Examples

- The large-scale power system design
- Vehicle routing problems
- Genetic identification
- Inverse chemical kinetics problem

Contents

Large-scale Global Optimization Problems-Definition

Complex Optimization Problems

- To effectively solve the large-scale global optimization problem, we normally do not search in the original search space.
- According to the correlation between variables, the problem can be separable, partially separable or non-separable.

Fully separable function

$f(\boldsymbol{x})$ is a **fully separable** function iff (if and only if)

$$\min / \max f(\boldsymbol{x}) = (f(x_1), f(x_2), \cdots, f(x_n)) \tag{1}$$

where $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ is a D-dimensional decision vector.

Large-scale Global Optimization Problems-Definition

Complex Optimization Problems

- To effectively solve the large-scale global optimization problem, we normally do not search in the original search space.
- According to the correlation between variables, the problem can be separable, partially separable or non-separable.

Partially separable function

$f(\boldsymbol{x})$ is a **partially separable** function with m independent subcomponents iff

$$\min / \max f(\boldsymbol{x}) = \left(f\left(\boldsymbol{x_1}, \cdots\right), \cdots, f\left(\cdots, \boldsymbol{x_m}\right)\right) \tag{2}$$

where $\boldsymbol{x_1}, \cdots, \boldsymbol{x_m}$ are disjoint sub-vectors of $\boldsymbol{x}$ and $2 \leq m < n$.

Large-scale Global Optimization Problems-Definition

Complex Optimization Problems

- To effectively solve the large-scale global optimization problem, we normally do not search in the original search space.
- According to the correlation between variables, the problem can be separable, partially separable or non-separable.

Fully non-separable function

$f(x)$ is a **fully non-separable** function, if every pair of its decision variables interacts with each other.

Large-scale Global Optimization Problems-Definition

Complex Optimization Problems

- To effectively solve the large-scale global optimization problem, we normally do not search in the original search space.
- According to the correlation between variables, the problem can be separable, partially separable or non-separable.

Partially additively separable function

$f(\boldsymbol{x})$ is a **partially additively separable** function iff

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} f_i\left(\boldsymbol{x_i}\right) \tag{3}$$

where $\boldsymbol{x_i}$ are mutually exclusive decision vectors of f_i ; m is the number of separable subcomponents.

Large-scale Global Optimization Problems-Difficulties

Difficulties

Non-separable and **overlapping** functions are the most difficult to solve:

- The search space generally increases exponentially.
- The algorithm is easy to fall into local optima.
- The objective function generally has the characteristics of nonlinear, non-convex, multi-modal and non-differentiable.
- Variables are partially separable or completely non-separable.

It is also challenging for EAs to explore the entire search space effectively.

Difficulties

Non-separable and **overlapping** functions are the most difficult to solve:

- The search space generally increases exponentially.
- The algorithm is easy to fall into local optima.
- The objective function generally has the characteristics of nonlinear, non-convex, multi-modal and non-differentiable.
- Variables are partially separable or completely non-separable.

It is also challenging for EAs to explore the entire search space effectively.

Two branches of metaheuristics

- Space decomposition methods (co-evolution).
- Non-decomposition-based methods.

Co-evolution Methods

Methods Mechanism

- Decompose the large-scale global optimization problem into several low-dimensional sub-problems.
- Use the divide-and-conquer method to solve the large-scale optimization problem.

Co-evolution Methods

Methods Mechanism

- Decompose the large-scale global optimization problem into several low-dimensional sub-problems.
- Use the divide-and-conquer method to solve the large-scale optimization problem.

Challenge

- The interaction between variables in indivisible problems.
- It will have a great influence on the optimization efficiency of the algorithm.

Co-evolution Methods

Methods Mechanism

- Decompose the large-scale global optimization problem into several low-dimensional sub-problems.
- Use the divide-and-conquer method to solve the large-scale optimization problem.

Challenge

- The interaction between variables in indivisible problems.
- It will have a great influence on the optimization efficiency of the algorithm.

Solution

Most decomposition methods try to identify the interacting variables and assign them to the same in the sub-question.

Co-operative co-evolutionary (CC) approach

Potter and De Jong proposed the co-operative co-evolutionary (CC) approach in 1994 to improve the performance of GAs.

Co-operative co-evolutionary (CC) approach

Potter and De Jong proposed the co-operative co-evolutionary (CC) approach in 1994 to improve the performance of GAs.

The framework of classic CC algorithm

- Problem decomposition.
- Subcomponent optimization.
- Cooperative combination.

Co-operative co-evolutionary (CC) approach

Potter and De Jong proposed the co-operative co-evolutionary (CC) approach in 1994 to improve the performance of GAs.

The framework of classic CC algorithm

- Problem decomposition.
- Subcomponent optimization.
- Cooperative combination.

Categories

Based on different grouping strategies, the co-evolution algorithms used to solve LSGO problems are mainly divided into two categories: static grouping methods and dynamic grouping methods.

Limitations of CC algorithm

- Being tested only for the maximum dimension of 30.
- Being less effective than classical GAs in non-separable problems.

Co-evolution Methods-Static Grouping

Limitations of CC algorithm

- Being tested only for the maximum dimension of 30.
- Being less effective than classical GAs in non-separable problems.

Solutions

- Liu et al. combined CC framework with FEP (fast evolutionary programming) to solve the problem of 100–1000 dimensional continuous optimization.
- Scholars tried to combine the CC idea with swarm intelligence-based algorithms, such as CPSO-SK and CPSO-HK.

Co-evolution Methods-Static Grouping

Limitations of CC algorithm

- Being tested only for the maximum dimension of 30.
- Being less effective than classical GAs in non-separable problems.

Solutions

- Liu et al. combined CC framework with FEP (fast evolutionary programming) to solve the problem of 100–1000 dimensional continuous optimization.
- Scholars tried to combine the CC idea with swarm intelligence-based algorithms, such as CPSO-SK and CPSO-HK.

Limitations of Static grouping

Only effective in low-dimensional problems.

Relationships between static and dynamic Grouping

- They both try to detect the interacting relationship between variables and assign the variables that interacts each other to the same sub-component.
- In static grouping, the number of sub-components (k) is fixed, while in dynamic grouping, the structure of sub-components can be dynamically adjusted.

Relationships between static and dynamic Grouping

- They both try to detect the interacting relationship between variables and assign the variables that interacts each other to the same sub-component.
- In static grouping, the number of sub-components (k) is fixed, while in dynamic grouping, the structure of sub-components can be dynamically adjusted.

Categories

Dynamic grouping methods can be divided into two categories based on the substitution of variables in group components: random dynamic grouping and learning-based dynamic grouping.

1. Random dynamic grouping

DECC-G algorithm

Yang et al. proposed a DE-based co-evolution algorithm (DECC-G) using random grouping strategy to attempt to solve the non-decomposing LSGO problem of 500–1000 dimensions.

- Decomposed an n-dimensional target vector into multiple low-dimensional subcomponents, each of which is evolved by a self-adaptive DE with a neighborhood search algorithm.
- Proposed an adaptive weighting framework to further improve solutions, which assigns a weight to each of the subcomponents after each cycle.
- An optimization algorithm is used to optimize these weights, and the dimension of the optimization problem is much lower than before.

The framework of the DECC-G algorithm

Algorithm 1: DECC-G algorithm

while *termination criterion is not fulfilled* **do**

  Set $i = 0$;

  The n-dimensional object vector is randomly divided into m s-dimensional subcomponents;

  while $i < m$ **do**

    $i++$;

    Evolve the ith subcomponent with a certain EA;

  Assign a weight vector for each subcomponent;

  Optimize them via a certain EA;

Multilevel CC algorithm (MLCC) algorithm

To improve the performance of DECC-G, Yang et al. proposed a new multilevel CC algorithm (MLCC).

- A decomposer pool is used in MLCC to reduce the impact of the group size on the objective function evolution.
- Each decomposer specifies a group size, select the current decomposer for grouping objective vector based on the performance of decomposers in evolution process, and updates the decomposer.

The performance of MLCC algorithm

This method is more applicable to most real-word problems.

2. Learning-based Dynamic Grouping
Decomposing problems with interactions between variables requires prior knowledge of the problem.

- In such methods, the identification of interactions between variables is learned by the characteristic experience of the problems obtained before or during the optimization process.
- The purpose of this type methods is to increase the chance of of interacting variables being assigned to the same subcomponent.
- Therefore, some learning-based grouping methods come to play.

Co-evolution Methods-Dynamic Grouping

CCEA-AVP algorithm

Ray and Yao proposed a co-operative co-evolutionary algorithm based on correlation matrix based adaptive variable partitioning (CCEA-AVP).

- In the first several cycles of running, a subcomponent contains all the variables, so the evolution process is similar to the standard EA.
- In the following cycle, the decision variables are divided into several sub-components according to the value of the correlation between decision variables.
- Thus, the variables whose correlation coefficient is greater than a certain threshold are placed in the same sub-component.

CCEA-AVP with an adaptive grouping method

Inspired by CCEA-AVP, CCEA-AVP with an adaptive grouping method was proposed. However, the method based on correlation coefficient has a large amount of calculations, and it cannot identify the nonlinear dependence between variables.

CBCC algorithm

A co-evolutionary algorithm was proposed to allocate computing resources according to the contribution of each sub-component called contribution based cooperative co-evolution(CBCC).

The performance of CBCC algorithm

The results show that if there is an imbalance among the separable and non-separable parts of the fitness value in the LSGO problem, this method can significantly reduce the computational time.

Non-decomposition-based Methods

- The non-decomposed method mainly attempts to improve the performance of the standard meta heuristic algorithm to solve the LSGO problem.
- Such algorithms focus on defining new mutations, selection and crossover operators, hybridization, opposition-based learning, designing and using local search, sampling operators, or variable population size methods.