A New Multistart Algorithm

Marcel Hunting
AIMMS Optimization Specialist

Webinar, March 22, 2017
Motivation & Credits

> Presentation by John Chinneck about CCGO at INFORMS 2015

> CCGO was compared with Knitro multistart and AIMMS multistart

> Joined work with Benjamin Harrach
Overview

> Introduction: Global versus local optimization

> The AIMMS multistart algorithm visualized

> The new dynamic algorithm
  • Constraint Consensus Method
  • How to use it

> Experimental results

> Conclusions & remarks
Global versus local optimum
Convex problems…

…local solver will do
Basic multistart algorithm

Input: X & Y

0. Preprocess model

1. Randomly generate X starting points and select Y best from X

2. FOR all Y starting points DO:
   • Skip starting point if it belongs to a cluster
   • Solve NLP to obtain local solution
   • IF local solution belongs to a cluster THEN update cluster
     ELSE create new cluster

3. STOP if iteration or time limit exceeded; else GOTO 1.
Dynamic multistart algorithm

Input: X & Y

0. Preprocess model.

1. Randomly generate X starting points and select Y best from X

2. FOR all Y starting points DO:
   • Skip starting point if it belongs to a cluster
   • Solve NLP to obtain local solution
   • IF local solution belongs to a cluster THEN update cluster ELSE create new cluster

3. STOP if iteration or time limit exceeded; else GOTO 1.
Dynamic multistart algorithm

0. Preprocess model. Select X&Y. **Create sampling box.**

1. Randomly generate X **starting points** and select Y best from X

2. **FOR all Y starting points** DO:
   - Skip **starting point** if it belongs to a **cluster**
   - Solve NLP to obtain **local solution**
   - **IF local solution** belongs to a **cluster** THEN update **cluster**
     ELSE create new **cluster**

3. **STOP** if iteration or time limit exceeded; else GOTO 1.
Dynamic multistart algorithm

0. Preprocess model. Select X&Y. Create sampling box. **Apply CCM.**

1. Randomly generate X starting points and select Y best from X

2. FOR all Y starting points DO:
   - Skip starting point if it belongs to a **cluster**
   - Solve NLP to obtain local solution
   - IF local solution belongs to a **cluster** THEN update cluster
     ELSE create new **cluster**

3. STOP if iteration or time limit exceeded; else GOTO 1.
**Dynamic multistart algorithm**

0. Preprocess model. Select X&Y. Create sampling box. Apply CCM.

1. Randomly generate X **starting points** and select Y best from X
   **Iteration 3**: Restrict variable bounds if large infeasibilities

2. FOR all Y **starting points** DO:
   - Skip **starting point** if it belongs to a **cluster**
   - Solve NLP to obtain **local solution**
   - IF **local solution** belongs to a **cluster** THEN update **cluster**
   ELSE create new **cluster**

3. STOP if iteration or time limit exceeded; else GOTO 1.
Dynamic multistart algorithm

0. Preprocess model. Select X&Y. Create sampling box. Apply CCM.

1. Randomly generate X starting points and select Y best from X
   Iteration 3: Restrict variable bounds if large infeasibilities
   Iteration > 5: Apply CCM if no solution found

2. FOR all Y starting points DO:
   • Skip starting point if it belongs to a cluster
   • Solve NLP to obtain local solution
   • IF local solution belongs to a cluster THEN update cluster
     ELSE create new cluster

3. STOP if iteration or time limit exceeded; else GOTO 1.
Dynamic multistart algorithm

0. Preprocess model. Select X & Y. Create sampling box. Apply CCM.

1. Randomly generate X **starting points** and select Y best from X
   Iteration 3: Restrict variable bounds if large infeasibilities
   Iteration > 5: Apply CCM if no solution found

2. FOR all Y **starting points** DO:
   • Skip starting point if it belongs to a **cluster**
   • Solve NLP to obtain **local solution**
   • IF local solution belongs to a **cluster** THEN update cluster
     ELSE create new **cluster**

3. STOP if **Iteration ≥ 5 and Bayesian rule satisfied** or
   if time limit exceeded; else GOTO 1.
Constraint Consensus Method

> Developed by John Chinneck (et al.)

> Goal is to find an almost feasible solution

> Projection method: iteratively adjust point to reduce constraint violations

> Fast! In each iteration “only” the violation of each constraint is calculated

> Rapid initial progress but slow if feasible region is approached

> GMP::Instance::FindApproximatelyFeasibleSolution
How to use Multistart?

> Install system module Multi Start

> Generate myGMP := GMP::Instance::Generate( MP );

> Call DoMultiStart procedure in module:

  • Has two optional arguments for specifying “X” and “Y” (default 0)
  • MulStart::DoMultiStart( myGMP );
  • MulStart::DoMultiStart( myGMP, 0, 0 );
  • MulStart::DoMultiStart( myGMP, 10, 5 );
Experimental results - Setup

- Compare performance of dynamic MS, basic MS, and single solve

- Using AIMMS 4.32 (64-bit) with CONOPT as solver (MS: 1 thread)

- Time limit of 300 seconds

- 140 medium sized NLP models:
  - 91 models from Chinneck test set for CCGO (Princeton/CUTE library):
    - 45 models with linear constraints; 27 convex
    - 46 models with nonlinear constraints
  - 49 models from MINLPLib2 (of which 3 relaxed MINLP models)

- Intel(R) Core(TM) i7 CPU 860, 2.8 GHZ, 12.00 GB computer with 8 Cores using Windows 7
Results – Feasible solution
Results – Running times
Results – Good solution

Performance Profile 5% Gap

- dynamic MS
- basic MS
- single solve
Results – Convex models
Experimental results - Observations

> For two models the solution returned by “single solve” contained large infeasibilities (0.3 and 40).

> Largest infeasibility using MS was 8e-5.

> If the MS algorithm finds multiple optimal solutions then it returns the one with the smallest infeasibility.

> The dynamic MS algorithm found a solution for all Chinneck models except one (drcav3lq).

> The dynamic MS algorithm found a solution within 1% gap for all Chinneck models except two (drcav3lq, cresc132).
Conclusions & remarks

> Using multistart helps to find better solutions for nonlinear models

> The new dynamic multistart algorithm performs better than the old basic multistart algorithm

> Several minor improvements have been made from which also the basic multistart algorithm benefits

> KNITRO has its own parallel multistart algorithm
  - No clustering
  - No extra license requirements

> GMP-AOA (MINLP) can be used in combination with multistart
  - Install system module **Multi Start**
  - Set GMPOuterApprox::UseMultistart := 1;
References

> Chapter 17.2 in the AIMMS Language Reference: The AIMMS Multistart Algorithm.

