

次世代の AI を実現する LocalSolver8.5

MSI 株式会社

*宮崎 知明 MIYAZAKI Tomoaki

1. はじめに

コンピュータの飛躍的な性能向上により、ビッグデータの発達とともに、リアルタイムで意思決定、判断をすることが可能になり、空前の AI ブームを迎えようとしている。

AI による自動化は以下に大別できる。

- センサと連動した自動運転制御
(自動車の安全装置、プラント自動制御など)
- 大量の実績データを統計処理し、統計確率にもとづいたオペレーションの自動化
(事務処理の代替など)
- 人間の思考パターン的高速シミュレーション
(ゲーム、ディーププランニング等)

現状の AI は統計予測をベースとしたものやセンサ等による情報に対する対処方法が明確に決まっているのが基本であり、意思決定理論が確定していることが重要である。これに対して、新しい事象に対しても対応できるようにすることが次世代 AI である。

一言で言えば、意思決定モデルを作成し、最適化手法を組み込むことにより、実績にない事象に対しても、さらなる効率化、自動化を可能することである。

従来の最適化手法(数理計画法システム)は、解法に即したモデル化が必要であるだけでなく、データの規模が大きくなると実用的な意味で最適解をだすことができなかったのが現状である。

産業界では、大規模、広範囲な情報、リアルタイムな情報活用により、各種データ精度の向上と生産効率の向上に研究が進みつつある。現時点では、大規模データを使った統計解析とニューラルネットワーク(ディープラーニング)の AI による自動化が主流であるが、LocalSolver 等による大規模な最適化計算やシミュレーション計算を実現することにより、より現実的なリアルライフの問題を解くことができるようになる。と考える。

2. これからの最適化

これからの AI による最適化では、以下の要件が必要になると考えられる：

- 大規模な最適化問題への対応
- 目的関数、制約条件の非線形化
- 時間軸をもった即時性への対応

100 万以上の 0-1 整数変数をもつような大規模な数理最適化問題は、既存の数理計画法システム(MIP)や制約論理システム(CP)では、解探索で組合せ爆発が起こり、実用時間内に解くことは出来なかったが、LocalSolver では数分で実用的な答えを出すことができる。

コンピュータの高性能化(計算スピード、大規模記憶容量、通信速度)により今や AI(人口知能)が盛んに叫ばれ、コンピュータは人間を超えられるかとの議論も盛んになっている。コンピュータによる AI が人間の頭脳を超える時代が 2045 年に到達すると提唱されている(シンギュラリティ)。

3. LocalSolver の概要

LocalSolver は、仏ブイググループ内の最適化問題を研究する仏ブイグ社の最適化部門により、汎用化を試行し始めたのが起点である。

フランスの OR チーム(ソルボンヌ大学准教授 2 人及びブイグ社実践家 4 人)により、2007 年から 5 年間の研究開発プロジェクトで LocalSolver を実現した(国の支援)。

MIP(混合整数計画法)では解けない大規模組合せ最適化問題(800 万以上の 0-1 整数変数)を実用的に解くメタヒューリスティック解法(局所探索解法)をベースにシンプレックス法をも取り込んだ全く新しい汎用解法である。また、非線形制約関数、非線形目的関数まで拡張した問題(Mixed-variable non-convex programming)を解くことができ、現在も進化を続けている。

LocalSolver は、パラメータのチューニングを必要

とせずに、さまざまな最適化手法（ローカルおよび直接検索、制約の伝播と推論、線形および混合整数計画法、非線形計画法など）を組み合わせ最適化を行うことができる。

他の数学最適化ソフトウェアとは異なり、LocalSolver は単一の最適化手法に基づいておらず、独自の検索アプローチにより、最適化中にさまざまな最適化手法を動的にハイブリッド化している。LocalSolver は、局所探索法と直接探索法、制約伝播法と推論法、線形および混合整数計画法、非線形計画法を組み合わせ、最高の問題を解決する。

LocalSolver は強力なモデリング言語を持つ。LocalSolver プログラミング言語（LSP）は効率的なプログラミングスタイルを提供している。最新の関数型言語スタイルであり、動的だが強く自動化されている。暗黙の変数宣言、コンパクトなループ構文などであり、多くの関数は数学モデリングまたはプログラミングの両方に使用できる。

LSP の目標はプロトタイピング作業（信頼性）を実現するだけでなく、プログラミング努力をできるだけ減らす（効率）ことにある。結果として得られる LSP モデルは、既存のモデリング言語で書かれたものよりも冗長ではなく読みやすくなっている。

LocalSolver はフランスで開発が進められている次世代の最適化ソルバーであり、リアルライフの問題を現実的な時間内で解決することを目指したソフトウェアである。図 1. に LocalSolver の適用範囲を示す。

4. LocalSolver 8.5 の最新機能

LocalSolver 8.5は、既存のLP問題、MIP問題だけでなく、NLP問題をもカバーすることができる。

また、今までは、最適解の上界をもとめることは出来なかったが、前処理のモデル分析等で最適解の上界を求めることができるようになり、グローバルな意味で、最適化ソルバー（All-terrain & All-in-one solver）とよぶことが出来るようになった。

LocalSolver8.5の新機能を以下に示す：バージョン 8.5 は、2019 年 1 月末にリリースされた。

1) 大規模組み合わせ最適化の大幅機能アップ

セット&リストベースを使ったモデリングにより、ルーティング問題に関連するモデルでのパフォーマンスを大幅に向上。たとえば、LocalSolver 8.5 は、何千ものクライアントがサービスを提供する Vehicle Routing (CVRP) および Pickup&Delivery (PDP) の問題で世界最強の最適化ソリューションを提供する。

特に、ルーティングモデルの場合、CVRPLIB インスタンスの平均ギャップは、以前の 8.0 バージョンの 5% からこの新しい 8.5 バージョンの 1.6% に減少する。

また、2019 年度中に、厳密な時間ウィンドウを伴うすべてのルーティング問題と、厳しい運用上の制約をもつスケジューリング問題の最適化ソリューションで最高の性能をもつソルバーの開発に取り組んでいく。

LocalSolverによるセットベースのモデリング機能により、ジョブショップなどのスケジューリング問題の容易なモデリングと最適化を高速に実現することを目指している。

2) 従来と同等以上の下界値（上界値）を実現

LocalSolver 8.5 は、従来の凸型解領域の問題だけでなく、bool、int、または float の意思決定変数で表現される非線形、非凸型、非滑らかなモデルの下界値を計算するため、強い下限から恩恵を受けるようになる。これらの範囲は、革新的な分枝限定的な発見的方法と組み合わせられた非線形緩和技術に基づいている。

結果として、LocalSolver は最適性のギャップを計算するとともに、中小規模の問題に対しては最適性を証明することができるようになる。このことは、LocalSolver は汎用的な従来の Cplex、Xpress、Gurobi ソルバと完全に競合できるだけでなく、(MI) QP または (MI) NLP に関する QP ソルバーとも競合することができるといえる。

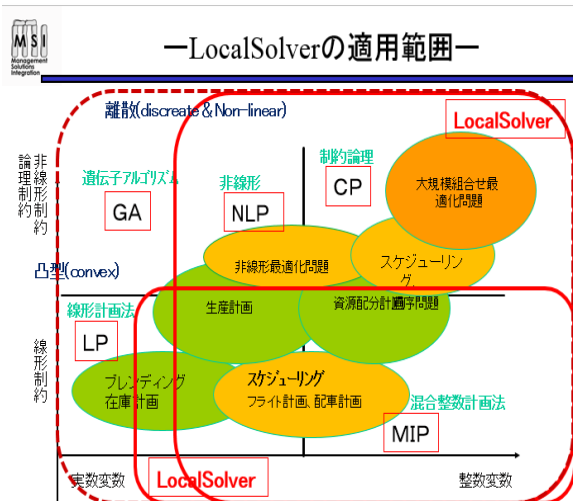


図 1. LocalSolver の範囲

さらに、内部最適化問題などの数値最適化問題を解決するために、LocalSolver がより優れた高速下限を生成する能力を強化するよう、内部ポイント法のような新しい数値アルゴリズムを開発中である。例としてはポートフォリオの最適化に加えて、エネルギーの分野にあるような離散的および連続的の両方の種類の変数に関わる問題も含まれている。また、セット・ベースのモデル、特にルーティングとスケジューリングから来るモデルの下限の計算についても、研究開発を進めている。

5. ベンチマーク結果

5.1 ベンチマーク巡回セールスマン

巡回セールスマン問題 (TSP) は次のように定義される。1 組の n 都市と各都市ペアの距離が与えられたら、各都市を 1 回だけ訪れる最小の長さの経路を見つける。

MIP モデルは、Dantzig、Fulkerson、および Johnson によって導入された、巡回セールスマン問題に対する正準混合整数計画法アプローチによって得られたものである。これは、ツアー内の 2 つの都市の連続を表す 2 次数の 2 進変数と、指数関数的な数の `subour` 除去制約で構成される。

LocalSolver モデルは、都市の順列を表すリスト変数が 1 つだけある。連続した都市間の距離は、'At' 演算子によって取得する。MIP モデルと比較して、このアプローチの利点は、それが簡単およびコンパクトであるということである。それはまた、はるかに良い結果を生み出すことを示している。

```
function model() {
```

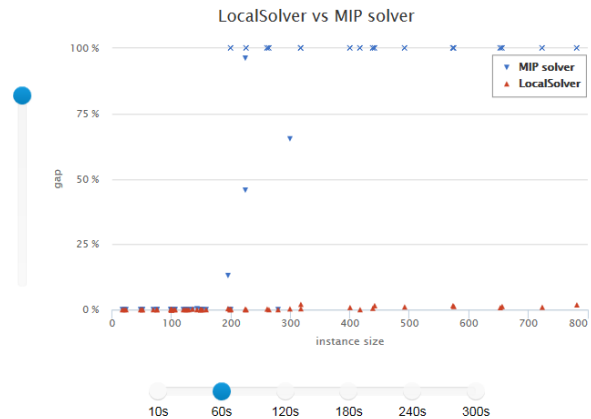
```
// A list variable: cities[i] is the
// index of the ith city in the tour
cities <- list(nbCities);

// All cities must be visited
constraint count(cities) == nbCities;

// Minimize the total distance
obj <- sum(1..nbCities-1, i => dis-
distanceWeight[cities[i-1]][cities[i]]
+ distanceWeight[cities[nbCities-
1]][cities[0]]);

    minimize obj;
}
```

このベンチマークでは、TSPLib の 144 の対称インスタンスについて、LocalSolver とその競合他社が異なる解決時間で得た結果を比較する。これらの問題はすべて既知の最適条件を持っているので、測定基準は最適性に対する相対的なギャップとなる。



5.2 二次割当問題 (QAP)

二次割り当て問題 (QAP) は、現実の問題に基づく基本的なオペレーショナルリサーチ問題です。 n 施設のセットと n 場所のセットがあり、距離は場所のペアごとに指定され、重みまたはフローは施設のペアごとに指定される (たとえば、2 つの施設間で輸送される物資の量)。問題は、距離の合計に対応するフローを掛けたものを最小にすることを目的として、すべての施設を異なる場所に割り当てることである。この問題の主な困難は、費用関数が二次的性質となる。

LocalSolver モデルは、ツアーの例で説明したように、QAP の最も単純な LocalSolver モデルは、工場の順列を表す単一のリスト変数に基づいている。リストの最初の項目は、最初の都市に割り当てられている工場の ID などに対応する。その場合、目的関数は単純明快で、完全モデルは 4 行しかかかりません。

```
function model() {

    x <- list(n);

    constraint count(x) == n;

    obj <- sum[i in 0..n-1][j in
0..n-1] ( distance[i][j] *
flow[x[i]][x[j]] );
}
```

```
minimize obj;
```

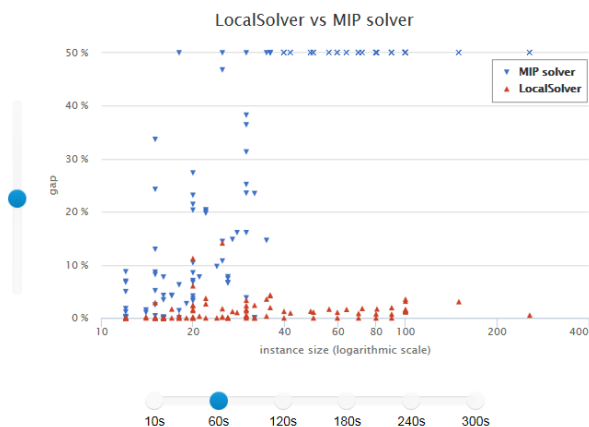
```
}
```

比較は、LocalSolver 6.0 と最先端の市販の MIP ソルバとの間で行われ、どちらも特別な設定や調整は必要ない。LocalSolver に使用されるモデルは上記の 4 行モデルで、MIP モデルは標準 2 次 (MIQP) 定式化である。

以下のチャートは、水平対数目盛のインスタンスサイズと垂直軸の既知の最適解との差を使って結果を示す。ソルバーが制限時間内に実行可能な解に到達できなかった場合、それは 'X' としてプロットされる。スライダーを使用してさまざまな制限時間を選択し、それに応じてチャートを更新することができる。

データは、QAP ライブラリからのものである。文献では、これらのインスタンスに対する解または下限を計算するために様々な技術が適用されており、そのうちのいくつかはいくつかのインスタンスの特別な構造を利用するかまたはランタイムの日数と共にスーパーコンピューティングクラスタを使用している。

このベンチマークでは、LocalSolver とその競合会社によって得られた結果を、デフォルト設定と限られた実行時間 (10 秒から 1 時間) で比較する。最適解が知られている 124 のインスタンスについて結果を報告し、この最適解とのギャップを以下のベンチマークの測定基準として使用する。すべての実験は MacBook Pro (Intel Core i5 2.5GHz、6GB RAM) で実行される。



MIP ソルバーでは、たとえ小さなインスタンスであっても、指定された実行時間内に質の高いソリューションを見つけるのが困難である。たとえば、5 分後に chr15b (15 か所) などのギャップがまだ 30% を超えて

いる。さらに、MIQP モデルの $O(n^4)$ サイズのため、45 を超えるロケーションを持つインスタンスに対して実行可能な解は生成されない。

それどころか、LocalSolver は問題の大きさに関係なく実行可能な解決策を即座に生成し、すぐに高品質の解決策に到達する。10 秒以内に、ほとんどの場合 10% 未満の最適性ギャップが得られる。計算時間を 5 分に増やすと、90% のインスタンスでギャップが 2% 未満に減少する。すべてのインスタンスと制限時間に対して、LocalSolver は 99.5% のケースで MIP ソルバーよりも優れたソリューションを提供する。

6. おわりに

AI ソリューションをさらなる進化させるために、大規模データと局所探索による最適化計算を加えることで、リアルライフの大規模最適化問題に対して、実践的な汎用アプローチが実現できる時代になったと考える。「実学に役立つ OR」として、人間と機械の調和、学術分野の統合を実現して日本の産業界の再生の一助となれば幸いである。

参考文献

- 1) MSI 株式会社
「<http://msi-jp.com/localsolver/>」ホームページ
- 2) 宮崎知明、山本邦雄、藤村 茂、三竹春子
・「次世代の AI を実現する LocalSolver8.0」、スケジューリング・シンポジウム 2018 論文集
- 3) 山本邦雄、三竹治子、宮崎知明
・「次世代 AI 開発手法研究」、日本 OR 学会春季研究発表会予稿集、1/2 (2019)