強制約条件付き最適化問題の2段階探索

余俊[†], 高木英行[††]

九州大学大学院芸術工学府[†], 九州大学大学院芸術工学研究院[††]

1　はじめに

単純で，微分のような空間情報を多く使わず，並列性に適しているなどの優れた特性がある進化計算は今や研究にはなくてはならないツールになっている．一方，実問題はどんどん複雑になってきており数理的手法だけでは設計が困難になってきているため，新幹線N700系先頭車両の形状設計のような進化計算の産業応用も進んでいる．マツダの車両設計もその一つで，その設計過程で得られたデータを基に実問題のベンチマーク問題として公開された[1)]．

本論文の目的は，進化計算が有効解を見つけることが難しい程の強い制約条件を持つ最適化問題に対処するために，2段階で探索する手法を提案することである．第1段階探索では，fitness計算を行わずに提案する制約度の概念で制約条件を満たす解を進化計算探索する．これはexplorationに相当する．第2段階探索では，第1段階探索で得られた有効解の局所探索をfitnessに基づいて行う．局所探索領域数は，領域の個体数やfitness計算制約数などを基に決める．提案手法を複数車両設計問題[1)]を使って評価する．このベンチマーク問題は単目的最適化と多目的最適化の両方あるが，本論文では，3種類の車種の合計重量を最小にする単目的最適化のベンチマーク問題で評価する．

第2節ではマツダのベンチマーク問題の概要を説明し，第3節で強い制約条件付き最適化問題の解法として2段階探索法を提案する．第4節では実験条件を変えて提案手法をベンチマーク問題に適用して評価をする．その後，第5節と第6節で考察と結論を述べる．

Two-step approach for optimization problems with strong constraints

† Jun Yu (yujun@kyudai.jp)

†† Hideyuki Takagi (http://www.design.kyushu-u.ac.jp/~takagi/)

Graduate School of Design, Kyushu University (†)

Faculty of Design, Kyushu University (†)

2　複数車種の同時最適化問題

資源の枯渇や環境問題への関心の高まりと共に，車を軽く共通部品を多くするような設計が求められる．このような車種開発の過程で得られたデータを基にマツダは，3車種を同時に設計し軽量で共通部品を増やすことを最適化目的とする実用的なベンチマーク問題を設計し公開した．

複数車両構造の設計問題は次のように定式化される．

設計変数

$$x_{SUV} = (x_1, x_2, ..., x_d)^T, d = 74$$
$$x_{CDW} = (x_{1+d}, x_{2+d}, ..., x_{2d})^T, d = 74$$
$$x_{C5H} = (x_{1+2d}, x_{2+2d}, ..., x_{3d})^T, d = 74$$

最小化

$$f_1 = M(x_{SUV}) + M(x_{CDW}) + M(x_{C5H})$$

最大化

$f_2 =$ 共通の厚さの部品数

制約条件

$$g_j(x_{SUV}) \geq 0,$$
$$g_{j+P}(x_{CDW}) \geq 0,$$
$$g_{j+2P}(x_{C5H}) \geq 0,\ j = 1, 2, ..., P,\ P = 18$$
$$\text{SUV: } x_i^L \leq x_i \leq x_i^U,$$
$$\text{CDW: } x_{i+d}^L \leq x_{i+d} \leq x_{i+d}^U,$$
$$\text{C5H: } x_{i+2d}^L \leq x_{i+2d} \leq x_{i+2d}^U,\ i = 1, 2, ..., d$$

ここで，dとPは，各々各車両の設計変数と制約条件の数である．x_k^Lとx_k^Uは，各々k番目変数の下限と上限である $(k = 1, 2, ..., 3d)$．f_1とf_2は，各々3車種の総重量を最小化する目的と，共通部品数を最大化する目的である．詳細は文献[1)]に詳しい．本論文での提案手法評価には，単目的最適化としてのf_1のみを扱う．

3　提案手法

強い制約付き最適化問題対策として2段階アプローチを提案する．進化計算はfitnessに基づいてより良い解探索を行うが，fitnessが良い解の分布と制約領域とは無関係であり，探索されたfitness

の良い解が制約条件のために採用できないことが多々ある．例えばFig. 1のように制約条件が厳しい場合，通常の進化計算探索をしても多くの個体は致死個体となり，不足する個体は毎世代乱数初期化で補うということもあり得る．

Fig. 1　強い制約条件付きの最適化問題の例．灰色部分は制約条件を満たす解が存在しない致死領域で，白色部分の解が全制約を満たす有効解領域．

この問題解決のため，提案手法は，第1段階で制約条件を満たす領域探索に専念し，第2段階でこの領域内での局所探索を行う．第1段階では，個体の質（fitness）に関係なく制約条件を満たす解をできるだけ探索する．複数の制約条件がある場合，全制約を満たす解が最適化探索に利用できる個体になるが，そのような解が存在する領域を探し出すことが第1段階の目的である．この際，以下で定義する「制約度」の概念を導入し，この制約度をfitness代わりにして全制約条件を満たす個体分布を進化計算探索する．

$$\text{制約度} = \frac{\text{制約条件を満たした制約数}}{\text{制約条件総数}}$$

第1段階で制約条件を満たす個体を十分収集するが，これらは質（fitness）を考慮していない．第2段階では，第1段階の個体を初期値にして質探索を行う．第1段階で十二分な個体数が得られた場合の個体選別方式は色々考えられる．fitnessの上位個体を選別する方法（次節の実験2），広い領域から選抜するためにランダム選択をする方法（次節の実験3），クラスタリングを行ってできるだけ多くの局所領域から個体を選別する方法等である．強い制約条件付き最適化問題ではFig. 1の白色領域のように解存在領域が狭いので，個体抽出をした後の第2段階では，局所探索に重点を置き，制約度は使わずにfitnessを探索指標にして探索を行う．全制約が満たされ，かつ，子個体fitnessが親個体を上回る時のみ親個体が子個体に書き換えられる．こうすることで，全制約条件を満たすよりfitnessの高い解のみが進化探索に使われる．

4　評価実験

マツダの単目的ベンチマーク[1)]を用い，複数の実験条件で提案2段階手法を評価する．提案法の第1段階の探索と第2段階の探索では異なる探索を行うので，異なる特性の進化計算をそれぞれで用いる．第1段階の広域探索には差分進化を，第2段階の局所探索には改良型花火アルゴリズム（enhanced firework algorithm (EFWA) [3)]）を用いる．最大fitness計算回数3万回までの実験を21施行行う．Table 1と2に，差分進化とEFWAの実験条件を示す．EFWAの突然変異演算を省いていることと，次世代候補選択は第3節で述べた全制約条件を満たし，かつ，fitnessが高い子個体を選択していることに注意されたい．Table 2の記号は文献[2, 3)]を参照のこと．

Table 1　差分進化パラメータの実験条件

実験2と3の個体数	50 および 400
scale factor F	0.8
交差率	0.9
DE 演算	DE/rand/1/bin

Table 2　花火アルゴリズムパラメータの実験条件

実験2と3の花火数	10
スパーク数m	50
A_{init} と A_{final}	0.2 and 0.05
定数パラメータ	$a = 0.04$ $b = 0.8$
最大振幅 A_{max}	0.5

実験1: 第1段階と第2段階の双方でEFWAのみ用い，第1段階では制約度を，第2段階ではfitnessを評価指標に用いる．

実験2: 第1段階では，制約度を評価値して必要数の解が得られるまで差分進化探索を行う．次に第2段階ではfitnessの上位t%の個体を使ってEFWAが探索を行う．残りの個体は捨てる．本実験では上位20%の個体を用いる．

実験3: 第1段階では，制約度を評価値して差分進化ですべての制約条件を満たす（制約度=1.0）個体を多く探索し，得られた多くの個体の中からランダムに必要個体数を選択して第2段階で用いる．ランダムに選択する個体数はTable 2の花火数である．

複数車種設計問題[1)]の単目的最適化を扱う場合，各車種の制約条件は他の車種に影響しない独立制約なので，各車種を個別に最適化する．Table 3, 4, 5は，3種類の車種毎に21試行各々で得られた三つの実験の最適解である．個別に最適化した車種3台を組み合わせた最終結果をTable 6に示す．

Table 3　Car 1：3実験21試行で3万回fitness計算までの間に得られた最適解．

試行No.	実験1	実験2	実験3
1	0.870051	0.874046	0.87435
2	0.856856	0.874046	0.868895
3	0.870436	0.874046	0.868895
4	0.872011	0.872327	0.868895
5	0.861202	0.863777	0.868895
6	0.868388	0.863777	0.868895
7	0.880849	0.863777	0.868895
8	0.879225	0.863777	0.868895
9	0.883587	0.863777	0.868895
10	0.860055	0.863777	0.868895
11	0.87572	0.863777	0.868895
12	0.871239	0.863777	0.868895
13	0.875408	0.863777	0.868895
14	0.878747	0.863777	0.868895
15	0.872412	0.863777	0.868895
16	0.881184	0.863777	0.868895
17	0.874638	0.863777	0.868895
18	0.863853	0.863777	0.868895
19	0.877931	0.863777	0.868895
20	0.880227	0.863777	0.868895
21	0.882249	0.863777	0.868895

5　考察

第1の議論点は，explorationとexploitationに相当する役割を2段階探索に分ける今回の提案手法の有効性である．提案手法は，第1段階では個体の良し悪し（fitness）に関係なく強い制約条件を満たす解をできるだけ多く探し出すことを目的とし，第2段階では，第1段階で得られた個体を初期値としてfitnessの良い個体を局所探索する．第1段階探索と第2段階探索ではどのような進化計算アルゴリズムを用いても構わない．

この提案方法の第1段階探索がないとほとんどの初期個体，あるいは，生成子個体が致死個体になってしまうため，進化計算自体が適用困難になってしまう．今回の複数車種設計ベンチマーク問題の場合，74次元の探索空間にランダムに100万個のサンプル点をバラまいたところ，致死個体にならなった個体数は，car 1 , car 2, car 3の各問題で631個体，28個体，3217個体であった．致死個体にならない領域がいかに狭いか，いかに制約条件が厳しいかを示している．第4節の評価実験では第1段階で差分進化が有効個体5000個を探し出した後に，第2段階のfitness探索を行っており，提案の第1段階探索がないと強い制約条件下での探索がいかに困難であるかが分かる．

Table 4　Car 2：3実験21試行で3万回fitness計算までの間に得られた最適解．

試行No.	実験1	実験2	実験3
1	0.916249	0.921679	0.87435
2	0.914693	0.920823	0.91716
3	0.90858	0.919088	0.91716
4	0.918082	0.919088	0.91473
5	0.92233	0.919088	0.91473
6	0.916316	0.911218	0.91473
7	0.918634	0.911218	0.91473
8	0.918744	0.911218	0.91473
9	0.919077	0.911218	0.91473
10	0.918314	0.911218	0.91473
11	0.928213	0.911218	0.91227
12	0.927203	0.911211	0.91227
13	0.919154	0.911211	0.91227
14	0.909379	0.911211	0.91227
15	0.919531	0.911211	0.91227
16	0.915614	0.911211	0.91227
17	0.924004	0.911211	0.91227
18	0.916255	0.911211	0.91227
19	0.913707	0.911211	0.91227
20	0.920072	0.911211	0.91227
21	0.922962	0.911211	0.91227

一般にfitness計算コストは制約条件のチェックコストに比較して各段に高いことが多い．そのような場合，計算コストの高いfitness探索をやみくもに行うべきではなく，本提案のように計算コストの少ない有効個体探索を多く行った上でfitness計算すべきであろう．我々に計算環境に余裕があれば，制約度景観が推定できる程に第1段階の有効個体探索を増やすことで，今まで見えなかった複数車種設計ベンチマーク問題の性質を解析できる可能性がある．その制約度景観が分かれば，fitness景観と合わせて見えなかった更に良い解領域が見つかる可能性がある．

第2の議論点は，大局的最適解が見つかっているかどうかである．前述したように，単目的最適化では，car 1 - 3問題で，100万個の乱数探索点中各々631，28，3217個体しか制約条件を満た

Table 5　Car 3：3実験21試行で3万回fitness計算までの間に得られた最適解.

試行No.	実験1	実験2	実験3
1	0.905423	0.913087	0.910882
2	0.899172	0.913087	0.910882
3	0.922545	0.913087	0.910882
4	0.912644	0.913087	0.908203
5	0.916126	0.913087	0.908203
6	0.910958	0.913087	0.908203
7	0.914717	0.913087	0.908203
8	0.905083	0.905116	0.908203
9	0.922712	0.905116	0.908203
10	0.910029	0.905116	0.908203
11	0.914043	0.905116	0.908203
12	0.920387	0.905116	0.908203
13	0.908736	0.905116	0.908203
14	0.907522	0.905116	0.908203
15	0.913341	0.905116	0.908203
16	0.901784	0.905116	0.908203
17	0.918484	0.905116	0.908203
18	0.913268	0.905116	0.908203
19	0.915769	0.905116	0.908203
20	0.910827	0.905116	0.908203
21	0.920192	0.905116	0.908203

Table 6　独立に探索したTable 3 - 5の設計車両3台を組み合わせたcompetition用の最軽量値.

試行No.	実験1	実験2	実験3
1	2.691723	2.708812	2.702393
2	2.670722	2.707956	2.696938
3	2.701562	2.706222	2.694508
4	2.702737	2.704503	2.691829
5	2.699657	2.695953	2.691829
6	2.695663	2.688083	2.691829
7	2.7142	2.688083	2.691829
8	2.703052	2.680112	2.691829
9	2.725376	2.680112	2.691829
10	2.688398	2.680112	2.691829
11	2.717975	2.680112	2.689369
12	2.71883	2.680105	2.689369
13	2.703298	2.680105	2.689369
14	2.695648	2.680105	2.689369
15	2.705284	2.680105	2.689369
16	2.698581	2.680105	2.689369
17	2.717126	2.680105	2.689369
18	2.693376	2.680105	2.689369
19	2.707407	2.680105	2.689369
20	2.711126	2.680105	2.689369
21	2.725403	2.680105	2.689369

せない程の強い制約条件を持つベンチマーク問題である．Table 6を見ると複数試行で同じ性能値に辿り着いているので，一見この値が大局的最適解ではないかとも思われるかもしれない．しかし，筆者らは第1段階探索で得る有効個体数をもっと増やすよう探索を続けてから第2段階探索を行うよう，実験を継続しているが，途中経過ではこの表の結果よりもさらに軽量な結果が得られ始めている．このことから，大局的最適解はまだまだ見つかっておらず，探索の工夫の余地があると言える．

6　結論

強い制約条件を持つ最適化問題の解法として，exploration と exploitationを個別に行う2段階探索方式を提案した．fitness計算コストが制約条件チェック計算コストに比べて非常に大きいタスクの場合，提案手法は計算コストを大きく増やすことなく効果的にfitnessの高い有効解を見つけ出すことが可能になる．また，この手法にはどのような進化計算アルゴリズムとも組み合わせることができるため，タスクの応じたアルゴリズム選択によって探索性能を上げることに影響を与えない．

今後の課題として，高次元問題の個体数などのように，適応パラメータ法も含み問題に適した進化計算パラメータの設定方法，制約条件の強さの判断と第1段階探索で探索する妥当な有効個体数の決定，今回導入した制約度に基づく，制約度景観の解析によるタスクの特性知見の獲得，有効個体分布領域を見つけ出すための第2段階探索で得られた有効個体のクラスタリング，がある．

謝辞

本研究はJSPS科学研究費（課題番号 JP15K00340）の助成を受けたものである．

参考文献

1) 小平剛央，釼持寛正，大山聖，立川智章,「応答曲面法を用いた複数車種の同時最適化ベンチマーク問題の提案」，進化計算学会論文誌, vol. 8, No. 1 pp. 11-21 (2017年).

2) Tan Y. and Zhu Y., “Fireworks Algorithm for Optimization,” In: Tan Y., Shi Y., Tan K.C. (eds) Advances in Swarm Intelligence. Int. Conf. on Swarm Intelligence (ICSI 2010). Lecture Notes in Computer Science, vol. 6145. Springer, Berlin, Heidelberg (2010).

3) Zheng, S., Janecek, A., and Tan, Y., “Enhanced Fireworks Algorithm,” IEEE Congress on Evolutionary Computation (CEC2013), pp. 2069–2077, Cancun, Mexico (June 20-23, 2013).