

## 3次元モデルの自動分解手法

穴井宏和 橋間正芳 洪明勲<sup>†</sup> 屋並仁史 金児純司 佐藤裕一  
(株)富士通研究所 <sup>†</sup>(株)富士通ハイパースoftwareテクノロジ

製品開発における組立・分解工程での部品間の衝突や保守性の不良の発見は、試作機が出来上がってからでは多大なコストの増大につながる。そのため CAD で作成した 3 次元モデル上で、仮想的に組立・分解工程検証のできるシステムの開発が望まれている。

我々は、近似凸分解に基づく経路探索手法を使い、3次元モデル上で部品の分解経路を自動生成できるシステムを実現した。本稿では、衝突無し分解経路の自動生成法とその効率化について実際の製品モデルにおいて検証した結果とともに報告する。

### Automatic generation of a disassemble path for 3-D models

Hirokazu ANAI, Masayoshi HASHIMA, Myunghoon HONG<sup>†</sup>, Hitoshi YANAMI,  
Junji KANEKO, and Yuichi SATO

Fujitsu Laboratories LTD <sup>†</sup>Fujitsu Hyper Software Technologies LTD

**Abstract** - In this paper we show an effective realization of automatic generation of a collision-free disassemble path of 3-D models, which is based on approximate convex decomposition of non-convex polyhedron and a probabilistic roadmap to search for collision-free paths. The system enables designers to construct an assemble/disassemble plan efficiently on a 3-D model of product before an actual prototype is completed, leading to an extreme improvement of product maintainability.

#### 1. はじめに

近年、ハード・ソフトの両面における 3 次元グラフィックス環境の進展に伴い、機械系 3 次元 CAD システムを利用した製品設計が普及している。計算機上で全ての設計評価を行うシミュレーション・ベースト・デザインを確立するためには、作成データを構造解析や電磁波解析等で利用したり、データを用いて組立て・分解シミュレーションやデザインレビューのできる設計環境を構築することが必要である。

その実現に向けて、本稿では、特に製品の分解性・保守性を評価するために必須となる自動分解機能について、そのアルゴリズムと実装及び実際の適用例について報告する。

本稿において考える問題は、部品の保守性検討 (assembly maintainability study) 問題である。具体的には「製品のある部品を、他の部分と衝突無く

取り出せるかどうか」を検討する問題である。以前はこのようなタスクは、物理的なモックアップを作成しその部品の分解経路を実際にモックアップを用いて見つけることを行っていた。これでは、衝突無し分解経路が見つかったとしてもアクセスに効果的な動きの列を把握することは容易ではない。また、組立・分解工程での部品間の衝突や保守性の不良の発見は、モックアップや試作機が出来上がってからでは多大なコストの増大につながる。したがって、最近ではできるかぎりモックアップ等を作成せずに 3 次元 CAD を用いて設計をするようになってきている。組立て・分解経路についても一度衝突無し経路が見つければ、その情報はコンピュータアニメーションによってシステムティックに保持することができ、後で容易に参照することができる。

しかしながら、この 3 次元モデルに基づく設計は、設計者に新たな大きな作業負担を課すことに

なる。3次元モデルをグラフィカル環境の中において他の部品との衝突を避けながら動かすことは、特に回転を伴うような場合に

- ・ 3Dモデルの2D表示の認識の問題、
- ・ マニピュレーションデバイスからの物理的なフィードバックの欠如、
- ・ 真の3Dディスプレイの欠如。

といった要因から、簡単なモデルの場合でもかなり大変な作業となっている。

これに対して、我々は、製品の3次元仮想モデル上で近似凸分解に基づく経路探索手法を使い、部品の分解経路を自動生成できるシステムを実現した。このシステムでは、製品の3次元モデルのデータを用いて、ユーザが指定した部品を他の部品と衝突すること無く、ユーザが指定した位置へ移動する経路を自動生成し、その結果をアニメーション表示することができる。また、衝突無し経路の生成に際して、ユーザがインタラクティブに経由位置を指定することもでき、得られた衝突無し経路についてもよりスムーズな経路を探索することも可能である。

このようなシステムを実現するために、3次元モデル(一般に非凸多面体)の凸分解に基づく高速干渉チェックと確率的ロードマップ法(PRM probabilistic roadmap)を組み合わせて衝突無し経路探索を行う方法を採用した。

その高速化のために、衝突なし経路生成に関係する部分に探索空間を絞り込んだ上で、衝突無し経路探索に適した「近似凸分解」を用いる。部品データの経路生成には不要な小さい穴を埋める機能も追加し凸分解をさらに高速化した。また、ユーザが途中経由位置を指定することで経路探索空間を複数の小問題に分割し全体の高速化を図った。

本稿では、上記衝突無し分解経路の自動生成法とその効率化について実際の製品モデルにおいて検証した結果とともに報告する

## 2. 3次元モデルの自動分解手法

本稿では、製品の分解性・保守性評価のための部品の自動分解機能を実現するためロボット運動計画(Robot Motion Planning)の手法<sup>[1]</sup>を用いる。ChangらがMotion Planningの手法を用いた方法を提案しているが<sup>[3]</sup>、複雑な形状モデルの場合には、形状の簡略化を用いることが提案されているものの、アドホックな手法しかなかった。

## 2.1 基礎アルゴリズム

我々の提案する方法は、確率的ロードマップ法<sup>[1,2]</sup>に基づく手法であり、部品同士の干渉チェックには、以下で説明する非凸3次元モデルの近似凸分解法を利用し、Configuration space(以後C-space)上での干渉チェックを用いる(Fig.3 詳細は<sup>[2]</sup>を参照されたし)。近似凸分解を用いることで、パラメータの調整によって、モデルを目的・問題に合わせて簡略化が可能となり、同時に凸形状同士の高速な干渉チェックを利用できる。これにより複雑な部品からなる問題の場合でも、パラメータの調整をすることでシステマティックに効率よく分解経路探索を生成できる。

## 2.2 近似凸分解

C-Space上でMotion Planningを行うためには、対象物と障害物の3次元モデルが凸形状である必要がある。そこで、前処理として、非凸部のある3次元モデルを凸形状に分解する。この凸分解手法としては、非凸部でモデルを切断・分割し、全ての非凸部がなくなるまで再帰的に繰り返す方法があるが、元の対象物/障害物の形状によっては切断回数が膨大になり、Motion Planningの計算量が増大してしまう。これに対して、大きな非凸部だけを分解する近似凸分解法が提案されており<sup>[4]</sup>、今回は、この手法を適用した。これは、3次元モデルの凸包構成と比較して非凸部の深さを測定し、しきい値以上の深さを持つ非凸部だけ切断・分離する方法である。Motion Planningでは部品表面の小さな凹凸は無視できるので、近似凸分解により必要な粗さで凸形状を得ることができる。

近似凸分解では、モデルの凸包構成、非凸部の深さ測定、一番深い非凸部でのモデル分割の3つの処理を繰り返していくが、この中で非凸部の深さ測定の処理は、凸包構成モデルと元のモデルの間で三角形ポリゴンの距離計算を行うため、ポリゴン数が増えると膨大な計算量となる。この問題に対して、グラフィックスボードによるハードウェア処理で高速化する方法を考案した。これは凸包構成モデルから元のモデルまでの最短距離をOpenGLのデプスマップ(Zバッファ)から測定するものである。具体的には、Fig.1-(a)のような非凸モデルに対して、Fig.1-(b)のような凸包構成モデルを作成する。この非凸モデルの各三角形ポリゴンを窓として、窓の中に見える元の形状をOpenGLで描画する。OpenGLでは手前に見える三

角形ポリゴンの距離をZバッファに格納するので、Zバッファから抽出した最大値が非凸部の深さとなる。この距離計算はグラフィックスボードでハード的に処理するので非常に高速である。

近似凸分解の例をFig.2に示す。なお、この手法では、分解対象が複数のモデルでも適用でき、機械製品等の3次元アセンブリモデルをまとめて近似凸分解することができる。

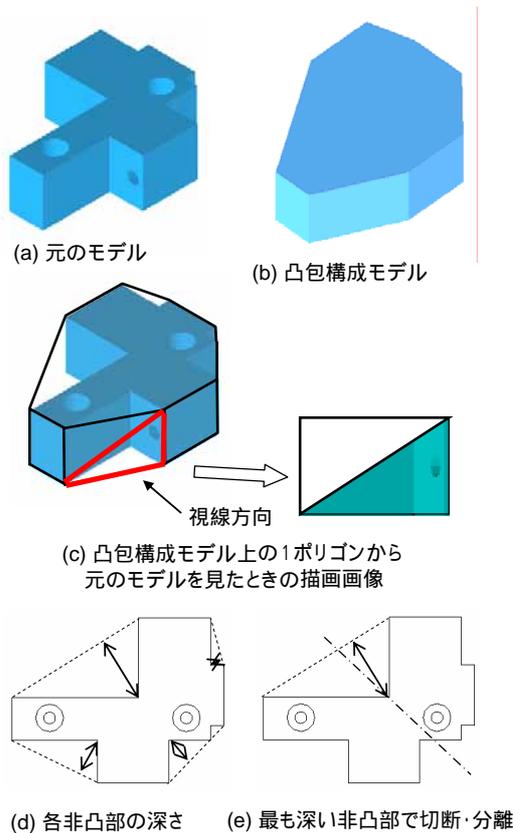


Fig.1 近似凸分解の概要

### 2.3 干渉チェック

凸形状同士の干渉チェックは、非常に簡単で凸モデルの頂点と面、面と面の接触条件の式をチェックする方法<sup>[1]</sup>を用いる。元の非凸モデルの干渉チェックはそれぞれの凸分解された凸成分同士の干渉チェックの結果を論理的に合わせればよい(詳細は<sup>[1]</sup>を参照)。

実際の干渉チェックは、まず、動かす部品と障害物の部品をそれぞれ包む直方体(bounding box)の座標で干渉の有無を大雑把にチェックする。bounding boxによるチェックで干渉が発見された場合に限って、上記凸形状同士の干渉チェック

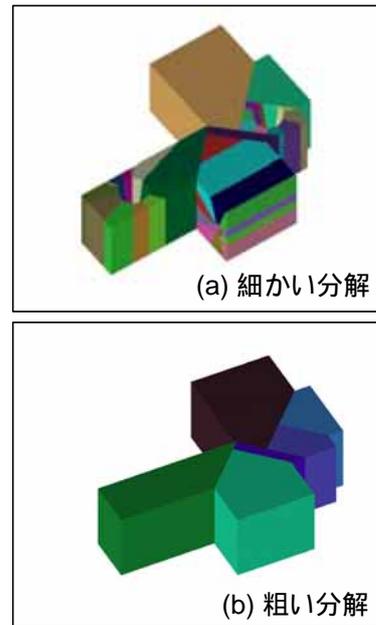


Fig.2 近似凸分解の例

を行うことで高速化を図っている。

### 2.4 確率的ロードマップ法による経路生成

C-spaceのどの点を選んで干渉チェックを行ってロードマップを構成していくのか、その戦略が大きく経路生成の効率化に影響する。確率的な手法に対してもこれまで多くの戦略が考えられてきているが、今回我々は基本的にはランダムにC-spaceの点を選んでいくが、bounding boxの情報を利用して探索空間をできるだけ小さくしている。候補点を1つずつ干渉チェックしながら衝突無しの候補点(milestone)からロードマップを構成して、衝突無し経路を探索している。

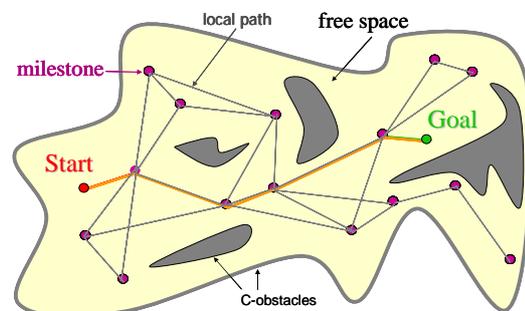


Fig.3 C-space とロードマップ法

この部分の計算時間は、今回試した例において

は全体の計算時間に対し非常に小さく、これだけの効率化でも十分に高速であったが、さらなる効率化には問題の特徴に対応したさまざまな戦略の実装が重要である。

### 3. 自動分解手法の高速化

#### 3.1 3次元モデルの限定・簡略化

我々の方法において支配的な部分は、凸分解である。特に複雑な形状では、非凸部の数が多く、近似凸分解の処理に時間がかかる。そこで、分解する必要のない非凸部を探索空間から事前に削除することによって、近似凸分解を高速化した。具体的な手続きは以下のとおりである。

まず、ユーザが GUI で探索空間を Motion Planning に関係するところだけに (Box で) 指定できるようにした。以後凸分解は Box 内の部品・

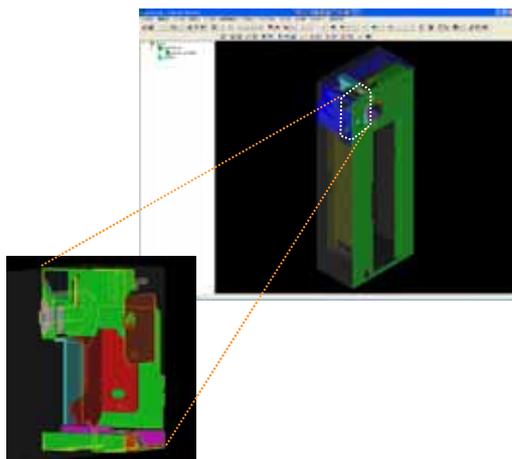


Fig.4 Box による探索空間限定

部分だけに対して実行される (Fig.4)。

次に、小さな穴埋めと近似凸分解の組み合わせである。機械部品にはネジ穴やコネクタ穴等の多くの小穴があり、これらは通常、Motion Planning には関係ないが、近似凸分解では非凸部として抽出され、その分解に無駄な時間がかかる場合がある。このような穴形状は三角形ポリゴンのトポロジー情報を使って検出することができ、穴の面積をしきい値判定して、小さな穴だけを埋めた形状を生成する (Fig.5)。

#### 3.2 ユーザとの協調機能

人間の認識能力を有効に活かすことで、衝突

無し経路探索を劇的に効率化することが可能となる。その1つとして、始点と終点以外に、経由点を GUI を使ってユーザが指定する機能を作成した。指定する C-space 上の点は、事前に干渉チェックが行われ、干渉なしの場合にのみ経由点として指定可能となる。これにより探索空間をさらに絞り込むことができ、不必要な空間を探索することも避けることができる。

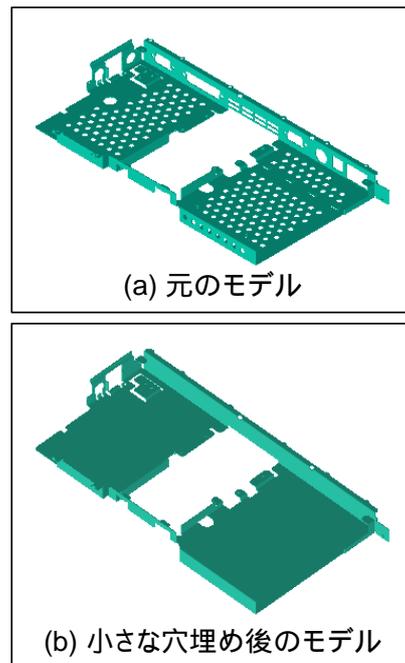


Fig.5 穴埋め処理

### 4. 適用結果と議論

本稿で紹介したアルゴリズムは、現在VPS<sup>[5,6,8]</sup>の拡張機能として実装している。試作結果について示す。まず、シミュレーション時間について計測した結果、CPU: Intel Xeon 3.2GHzのPCで、実際の製品モデル (Fig.4,6) の例の場合に、衝突無し経路を計算する時間は68.3secである。この例を実際のVPS熟練ユーザが画面上のマウス操作を通じて動的な干渉チェック機能を利用し衝突無し経路作成した場合に1時間程度を要した作業であったことから、従来の設計に比べ、設計工数を格段に短縮することができ、開発コストの削減にも貢献できることがわかった。

また、本システムをいくつかの例に適用した結果、Fig.4レベルの複雑さの例では開発早期に質の高い衝突無し経路生成および保守性評価を行うことが可能となり、無駄な試作を削減すること

ができることがわかった。実際の設計現場では、比較的単純な部品の場合にこのような分解経路を作成する必要が数多くあり、それらを全て手作業で設計者が行うことは大変に工数のかかる作業であり、今回のシステムはそのような設計工数を大幅に削減する効果があると期待できる。

今回構築したシステムの特徴を明確にするため、非凸形状体に凸分解を施さない干渉チェック法を用いた経路生成との性能比較を行った。この性能比較で採用した凸分解をしない干渉チェックは、RAPID<sup>[9]</sup>ベースの高速干渉チェック法である。(RAPIDの干渉チェックは、OBBTreeという階層的構造をもつbounding boxを用いたモデル間の距離計算無しの方法である<sup>[9,10]</sup>。)両者を比較して言えることは、以下のとおりである。FA環境のような移動体、障害物とも単純な形状で近似凸分解の負荷が軽いと考えられる場合には、ここで提案した凸分解を用いる経路生成手法の方が高速であった。一方、複雑な形状の場合では、凸分解を施さない高速干渉チェック<sup>[7,9]</sup>を用いた手法の方が高速であった。

また、部品同士が干渉ぎりぎりを通る経路は、ロバスト性を欠き実用上好ましくない。ロバストな経路生成を行うためには、凸分解を利用して部品形状を相似的に膨らませマージンを持たせて経路生成する、あるいは、<sup>[7]</sup>のような最近点と距離が高速に計算できるアルゴリズムを使って最近点を連続的に追跡しながら経路生成をすること等が必要となる。このとき経路生成全体としての方法がどのような場合に有効であるか比較検証する必要がある。今後の検討課題である。

今回の自動分解経路生成を実現するために開発した要素技術である3次元モデルの近似凸分解や穴埋め機能は、モデルの簡略化の有効な手法であり、これら自体で3次元モデルを用いた設計において必要となる様々なシミュレーションの効率化に有用である。

## 5. おわりに

設計者の手作業によって行われ多大な工数を要していた分解経路生成に対して、3次元モデル上で自動生成可能で分解・組立て評価ができる環境を構築した。

本稿で提案したアルゴリズムは、特に、比較的部品形状が簡単で複雑な経路を生成したい場合に有効であることが確認できた。一層複雑なモデルを扱えるようにするには更なる効率化が必要

であり、ロバストな経路生成法とあわせて今後の課題となっている。

今後は、計算効率の改善を行うとともに、接触を伴うような経路生成やスナップフィット部品など、3次元的な部品の変形を扱うことができるシミュレーション方法も開発する予定である。

## 謝辞

本研究への貴重なアドバイスいただき、また、モデルデータを提供して下さった富士通(株)の関係者各位に感謝致します。

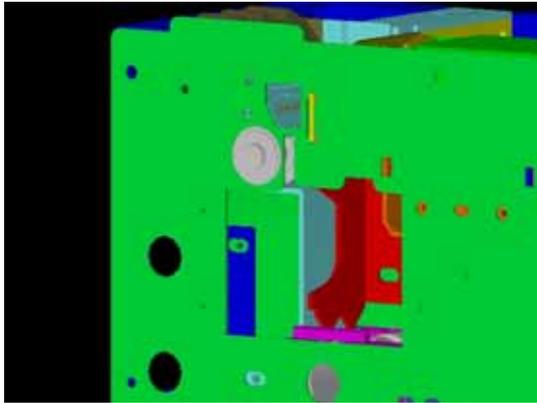
## 参考文献

- [1] J.C.Latombe. "Robot Motion Planning", Kluwer Academic Publishers, Boston, 1991.
- [2] L. E. Kavraki and J.-C. Latombe. "Probabilistic roadmaps for robot path planning : Current Approaches and Future Challenges", pp 33--53. John Wiley, West Sussex, England, 1998",
- [3] H.Chang and T-Y Li. "Assembly Maintainability Study with Motion Planning" IEEE Int'l Conf. on Robotics and Automation. pp. 1012-1019. 1995.
- [4] J. -M. Lien and N. M. Amato, "Approximate Convex Decomposition of Polygons", Proceedings of the 20-th Annual Symposium on Computational Geometry, pp.17-26, 2004.
- [5] M.Hashima, Y.Senta, and Y.Sato, "Design and Manufacturing Methodology for Mechanical Systems Using Virtual Product Simulator", ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC/CIE-34384, 2002.
- [6] 千田, 橋間, 佐藤, 3D CAD データを用いた組込み用ソフトウェア開発支援システムの構築 -第一報 システムの概要と基本設計-, 情報処理学会「グラフィックスとCAD」研究報告会, 109-1, pp.1-6, 2002
- [7] Y.Sato, M.Hirata, T.Maruyama, and Y.Arita, "Efficient Collision Detection using Fast Distance-Calculation Algorithms for Convex and Non-Convex Objects," IEEE International Conference on Robotics and Automation, pp. 771-778, 1996.
- [8] <http://salesgroup.fujitsu.com/plm/vps/>

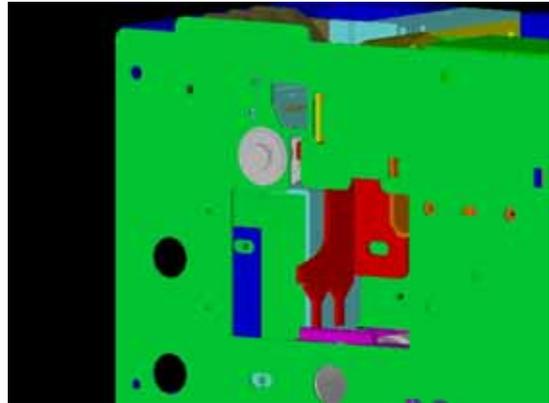
[9] S. Gottschalk, M. C. Lin and D. Manocha  
 "OBB-Tree: A Hierarchical Structure for Rapid  
 Interference Detection", In Proc. of Conference  
 on Computer Graphics, pp. 171-180, ACM,  
 August 4-9 1996

<http://www.cs.unc.edu/~geom/OBB/OBBT.html>

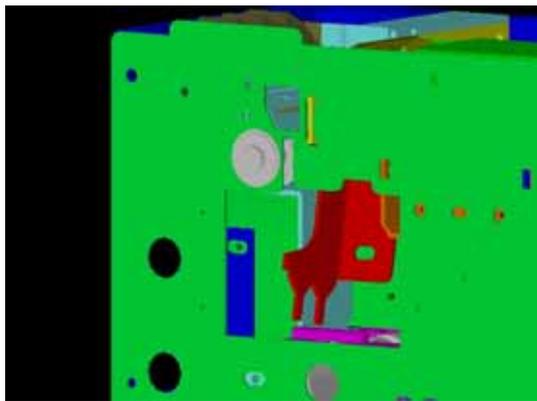
[10] RAPID --Robust and Accurate Polygon Interference



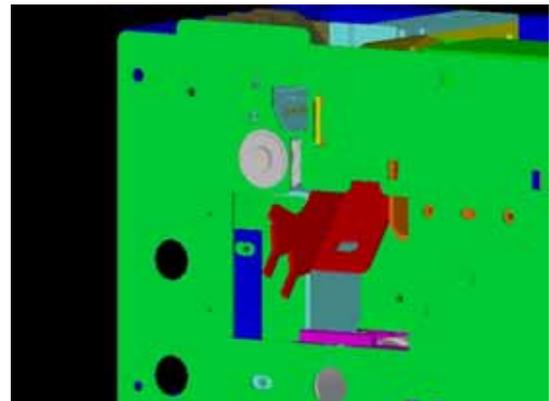
(a) start



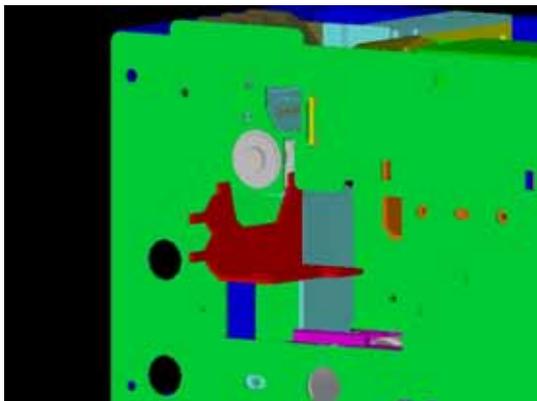
(b)



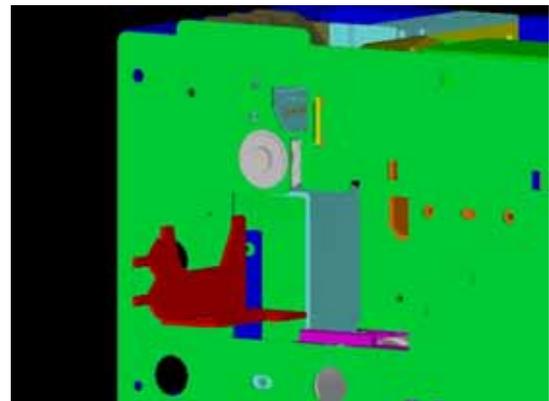
(c)



(d)



(e)



(f) goal

Fig.6 機械部品の衝突無し分解経路