

# GridRPCを用いた遺伝的アルゴリズムによる 構造物の位相最適化の検討

梶原 広輝<sup>†</sup> , 廣安 知之<sup>††</sup> , 三木 光範<sup>††</sup>

<sup>†</sup> 同志社大学大学院 <sup>††</sup> 同志社大学工学部

構造形態の最適化を行うためには、構造物の位相と形状を最適化する必要がある。本研究では、位相構造最適化に遺伝的アルゴリズム (Genetic Algorithm: GA) を用いることを検討する。GA を位相構造最適化に適用することで、従来の手法では創生できない、複数の制約条件を考慮した、より合理的な構造形態を創生できることを示す。GA を適用する上で問題となってくる多くの計算資源の確保には、GridRPC システム Ninf-G を利用することを検討する。近年の Grid 環境は、各拠点に分散する PC クラスタシステムを相互に接続した形態がとられており、PC クラスタ間を代表的な Grid ミドルウェアである Ninf-G で通信する。また、PC クラスタ内では、設計用大規模計算力学システム (ADVENTURE) の並列構造解析を利用することで、構造解析も並列実行する。PC クラスタ間を Ninf-G で通信し、PC クラスタ内を ADVENTURE システムで並列で構造解析することで、計算時間の大幅な削減が期待できる。

## Topology Structural Optimization using GA and GridRPC

Hiroki KAJIWARA<sup>†</sup> Tomoyuki HIROYASU<sup>††</sup> Mitsunori MIKI<sup>††</sup>

<sup>†</sup> Graduate School of Engineering, Doshisha University <sup>††</sup> Department of Knowledge Engineering,  
Doshisha University

When structural optimization is performed, it is necessary to optimize the topology and shape of the structures. In this research, Genetic Algorithm (GA) is applied for optimizing the topology of the structures. When the conventional methods are used, the constraints and objective function are limited. On the other hand, GA can deal with any constraints and objective function that are interested by the designers. On the other hand, GA needs the high calculation cost. To solve this problem, we examine to use GridRPC system Ninf-G to ensure huge calculation resources. The grid environments often consist of PC cluster systems that are located in distributed area. The communication between PC clusters is performed by Ninf-G; this is the typical Grid middleware. At the same time, analyzing structures is also performed in parallel using the ADVENTURE system. It is highly expected to reduce the high calculation time when the communicating between the PC cluster system is performed with Ninf-G and analyzing the structures in parallel with the ADVENTURE system in the PC cluster system.

### 1 はじめに

構造形態の最適化を行うためには、構造物の位相 (トポロジー) と形状を最適化する必要がある。一般的に構造物の位相最適化問題は、構造形態の重量や体積を最小化する位相を設計する問題として捉えられている。構造物の位相を最適化する代表的な手法として、均質化法<sup>1)</sup> や密度法<sup>2)</sup> といった数理計画法に基づく手法や、進化的構造最適化 (Evolutionary Structural Optimization: ESO)<sup>3)</sup>

やセル・オートマトン<sup>4)</sup> といったヒューリスティック手法に基づいた手法が挙げられる。先に述べた数理計画法に基づく手法は、局所的探索であるため大域的最適解を持つ多峰性の問題には不向きである。ヒューリスティック法に基づく手法は、単純なルールを設定するだけで任意の目的関数において大域的最適解を得ることが可能だが、複数の制約条件があるような問題には適用できないことがある。先に述べた ESO は応力値の小さい不要な

要素を序々に削除していく単純な手法であるため、許容応力や許容変位の制約を考慮することができない。構造物への荷重に対して、応力や変位が小さく体積が最小であるような合理的な構造形態を創生するためには、大域的探索が行え複数の制約条件と任意の目的関数を取り扱える手法が必要となる。

一方で、生物の進化と自然淘汰を工学的に模倣した最適化モデルとして遺伝的アルゴリズム (Genetic Algorithm: GA)<sup>5)</sup> がある。GA は複数の制約条件を持つ多峰性の強い最適化問題に対しても、多点探索により局所的最適解に陥ることなく、任意の目的関数において大域的最適解が得やすい手法とされている。そのため、許容応力や許容変位といった複数の制約条件がある構造物の位相最適化問題に適用することで、従来手法以上の効果が得られると考える。本研究では、構造物の位相最適化問題へ GA を適用する手法を対象としている。GA を位相最適化問題へ適用し良好な結果を得るためには、多くの世代数や個体数、評価計算回数が必要となり、その結果多くの計算時間が必要となる。より大規模・複雑な構造物を対象とする場合、1 回の評価に多くの計算時間が必要となる。そのため、位相最適化問題へ GA を適用する際には、評価の計算時間を短縮する仕組みが必要となる。

近年、広域ネットワーク上のあらゆる計算資源を仮想的に統合し利用する Grid の研究が盛んに行われている<sup>6)</sup>。特に科学技術計算の分野において、Grid 上で並列分散計算を行うためのミドルウェアである GridRPC の開発やその応用研究が盛んに行われている<sup>7)</sup>。GridRPC を用いることによって、最適化問題の多くの計算時間を必要とする解析処理を、広域ネットワーク上に分散する PC クラスタシステムで並列処理が行え、計算時間の大幅な削減が期待できる。

以上のことより、位相構造最適化問題に GA を適用する際に多くの計算量を占める解析処理を、代表的な GridRPC システムである Ninf-G<sup>8)</sup> を用いることで、計算時間の短縮を図ることを目的とする。本論文では、解析処理の並列分散処理を Ninf-G で行うための検討を行う。

## 2 遺伝的アルゴリズムによる位相構造最適化

本研究では、構造物への荷重に対する許容応力と許容変位の制約条件のもと、体積最小の構造レイアウトを創生することを目的とし、最適化手法に遺伝的アルゴリズム (Genetic Algorithm: GA)<sup>5)</sup> を用いる。本章では位相構造最適化問題へ GA を適用し構造レイアウトを創生する手順と考慮すべき事項について述べる。

### 2.1 位相構造最適化問題の解法手順

GA では、初期探索点として、個体の母集団を生成し、それぞれの個体には適合度が設定される。そして生成された母集団に対し、選択、交叉、突然変異といった遺伝的操作を繰り返し適用する。選択で適合度の高い個体が多く選ばれ、交叉で親個体の良質な形質を継承し、突然変異で個体の一部を変更することで多様性を維持し、良好な個体を増やすことにより、最適解を見つけ出す探索方法である。位相構造最適化問題へ GA を適用し構造レイアウトを創生する手順を以下に示す。

1. 初期個体の生成と初期化
2. 遺伝的操作 (選択, 交叉, 突然変異)
3. ESO の適用 (要素除去)
4. 構造解析 (各節点の変位と各要素の応力解析)
5. 制約条件判定および子個体の引き戻し
6. フィルタリング
7. 評価 (適合度計算)
8. 終了判定 (終了世代数まで上記 2~7 を繰り返す)

### 2.2 コード化と適合度計算

#### 2.2.1 コード化

本手法では、1 構造物を 1 個体として表現することで有限要素を遺伝子にコード化する。以下 Fig. 1 に示すように、個々の有限要素ごとに存在の有無を 0・1 ビットで表現し、各ビットに対応した 2 つの弾性係数の極端な差異を利用して構造レイアウトの位相を表現する。

#### 2.2.2 適合度計算

本研究では、許容応力と許容変位の制約条件のもと、体積最小化問題として構造レイアウトを創生する。式 (1) における第 1 項では、個体の持つ中実要素数を最小化する。第 2 項においては、最

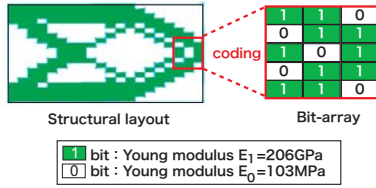


Fig. 1 Coding by the bit-array representation

大応力  $\sigma_{max}$  と許容応力  $\sigma_g$  との許容応力比に係数  $\zeta$  を乗じた項を加え、第 3 項においては最大変位  $\delta_{max}$  と許容変位  $\delta_g$  との許容変位比に係数  $\xi$  を乗じた項を加える。要素数が同等の個体間においては、構造物の最大応力と最大変位が許容応力と許容変位に対して余裕がある個体の適合度を高めるようにする。第 4 項においては、要素相当応力の分散値  $\sigma_v$  を  $\gamma$  でスケールリングし、係数  $\lambda$  を乗じた。一般的に構造物に加わる応力が分散されている場合と比較して、応力が集中している場合は構造物に亀裂が生じやすい。そのため要素相当応力の分散が小さい構造物ほど亀裂が生じにくく、適合度を高めるようにする。

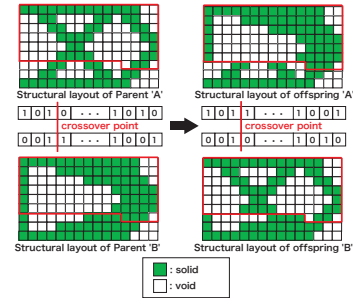
$$F = \sum_{i=1}^n x_i + \zeta \left( \frac{\sigma_{max}}{\sigma_g} \right) + \xi \left( \frac{\delta_{max}}{\delta_g} \right) + \lambda \left( \frac{\sigma_v}{\gamma} \right) \quad (1)$$

$$x_i \in \{0, 1\} \quad \text{Subject to: } \sigma_{max} < \sigma_g, \delta_{max} < \delta_g$$

### 2.3 構造形態を考慮した交叉

対象問題へ GA を適用する際には、親個体の良質な形質を子個体へ継承させる有効な交叉を考慮する必要がある。通常のビット GA における 1 点 (多点) 交叉をレイアウト最適化問題へ適用すると、Fig. 2(a) のように、構造形態とは無関係に交叉が行われる。

そこで、構造形態に基づいた交叉方法を検討する。本交叉は Fig. 2(b) に示すように 2 つの親個体 (親個体 'A' の構造レイアウトと親個体 'B' の構造レイアウト) の論理和と論理積から 2 つの子個体を生成する。子個体 'A' の構造レイアウトは、親個体 'A' と 'B' どちらかの要素に中実要素があれば、その要素は中実要素となる。子個体 'B' の構造レイアウトは、親個体 'A' と 'B' とともに中実要素があるときのみ、その要素は中実要素となる。これにより、有限要素数は多いが強度が強く最大応力・最大変位が小さい個体と、有限要素数は少ないが強度が弱く最大応力・最大変位が大きい個体の、2 種類の性質を持つ子個体が生成される。



(a) One point crossover  
(b) Crossover that considers structural form

Fig. 2 Comparison of structural layouts by crossover methods

## 2.4 ESO の導入と子個体の引き戻し

### 2.4.1 ESO の導入による解探索性能の向上

位相構造最適化の最も代表的な手法である、進化的構造最適化 (Evolutionary Structural Optimization: ESO) は、以下 Fig. 3 に示すように、通常の有限要素解析を繰り返しながら、削除率と呼ばれる閾値よりも応力値の小さい不要な部分を破棄する。この単純な過程を積み重ねることにより最適形態を求めるものである。構造最適化計算のための特別な工夫を要しないという点で拡張性と汎用性を持つ優れた手法であり、従来より多くの構造物の位相最適化問題に適用されてきた。

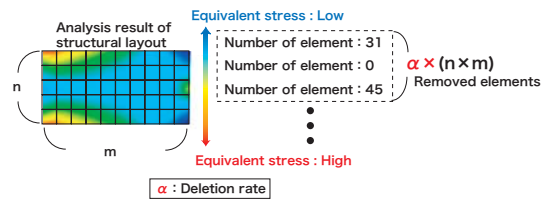


Fig. 3 ESO method

1 章でも述べた通り、一般的に GA では良好な解を得るためには多くの個体数や世代数が必要となり、最終的な解を求めるために多くの計算時間が

必要となり、進化の度合いが遅くなってしまふ。そこで、ESO の不要部分を削除するという概念を、2.1 節で示した GA の進化過程で導入することで収束が早まることが期待できる。また ESO は位相構造最適化に対して有効な手法であるため、ESO を GA の進化過程で適用することで、GA 単独の探索に比べ良好な解探索が行えることが期待できる。

#### 2.4.2 制約条件を外れた子個体の引き戻し

前述の ESO の導入、および GA の進化過程において、許容応力・許容変位の制約条件を外れた子個体に対しては、以下 Fig. 4 に示す子個体の引き戻しを適用する。

Fig. 4 において、親個体'A' から生成された子個体'a' が制約条件を満たせず実行可能領域から外れた場合、親個体'A' の設計変数と子個体'a' の設計変数を比較する。そして、子個体'a' のうち空疎要素でありかつ、親個体'A' のうち中実要素である要素を、中実要素にする。これにより、子個体は親個体と同等以上の中実要素を得ることで、その結果最大応力・最大変位は小さくなり、実行可能領域へ引き戻すことが可能となる。引き戻しにより子個体の有限要素数は親個体以上になってしまうが、前述したように個体の適合度は有限要素数と許容応力比、許容変位比、要素相当応力の分散によって決定するため、引き戻した子個体が次世代へ残る可能性は残ることとなる。

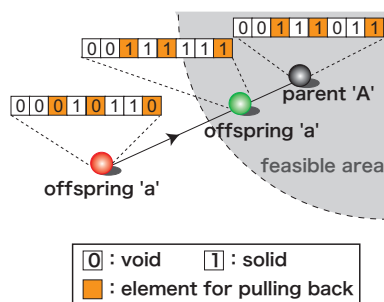


Fig. 4 Pulling back method

#### 2.5 構造レイアウトの創生結果

レイアウト最適化問題において、要素数 400 の 2 次元連続体の片持ち梁問題に対して、本手法を適用しその有効性を検証する。用いる手法は、本手法の ESO の概念を導入した GA (GA+ESO) と ESO の概念を導入しない GA (GA), 代表的な手法の ESO である。その際、GA+ESO においては削除

率を 0.0025, 0.0050, 0.0075, 0.0100 とし、ESO では削除率 0.01, 世代数 50 とした。

以下 Fig. 5 に最終的に得られた最終レイアウトを、Table 1 に最も良好な探索を示した GA+ESO (0.0025) と ESO との比較を示す。Fig. 5 より、本手法が従来の ESO では創生できない構造形態を創生していることが分かる。また Table 1 より、本手法が ESO では扱えない複数の制約条件を考慮した、より合理的な構造形態を創生していることが分かる。

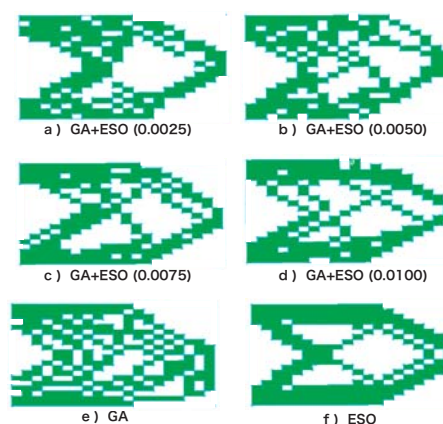


Fig. 5 Final structural layout

Table 1 Comparison between GA+ESO (0.0025) and ESO

	GA+ESO	ESO
Number of elements	176	200
Max stress (e+10)	3.096101	3.127697
Max displacement	6.743508	6.799343
Decentralization (e+19)	7.215970	7.435041
Fitness	178.955751	209.435041

### 3 GridRPC システム Ninf-G を用いた GA による位相構造最適化の検討

前章で述べた GA による位相構造最適化を、GridRPC システム Ninf-G を用いて並列分散化する検討を行う。

#### 3.1 現状の性能

GA による位相構造最適化では、全体の計算量のうち多くの部分を解析処理に要する。2.1 節において解析処理が必要となるのは、ステップ 4 の構造解析とステップ 5 の子個体の引き戻し後の構造解析のステップである。ステップ 4 においては、全

個体で毎世代定期的に構造解析が行われるが、ステップ5においては実行可能領域の制約条件を外れた子個体に対してのみ不定期に構造解析が行われる。位相構造最適化問題では、許容応力と許容変位に関する2つの非常に制約の厳しい制約条件が設定されており、探索序盤から制約条件付近の探索が行われるため、解析処理が頻繁に行われる。

2.5節で示した結果を得るためには、1個体を解析するために1CPU (Pentium4:2.8GHz) で約7.4秒となり、100個体×1000世代で100,000評価計算回数となり、少なくとも約8.6日はかかる。これに前述した子個体の引き戻し時にかかる解析時間と、GAの進化計算を含め、約12日間を要する。

### 3.2 ADVENTUREシステムによる 構造解析の並列化

本研究では位相構造の応力・変位解析に、設計用大規模計算力学システム開発プロジェクトで開発されている「設計用大規模計算力学システム：ADVENTURE」<sup>9)</sup>を用いている。ADVENTUREは、1千万~1億自由度級の大規模メッシュを用いて自然物や人工物を詳細にモデル化し、多様な並列分散計算環境のもとで固体の変形や熱、流体の流れ等の力学解析から、可視化や設計最適化までを行える汎用並列計算力学システムである。以下に示す特徴を有する。

- 大規模メッシュによる解析：  
1千万~1億自由度級の大規模メッシュを用いた構造、流体、熱の力学解析が可能。本研究では、構造解析を利用する。
- 超並列環境計算環境での並列処理：  
2000プロセッサの超並列計算環境においても高い並列効率を実現可能。
- 拡張性と保守性：  
7つの主要モジュールと12の追加モジュールの計19モジュールから構成されており、全モジュールで標準のI/Oライブラリーを採用。

以下 Fig. 6 に示すように、位相構造の解析処理にADVENTUREシステムを用いることで、1個体の構造解析をPCクラス内で複数CPUを利用した並列処理が可能となる。GAで得られた個体から位相情報をメッシュファイルに変換し、その後ADVENTUREシステムで解析を行う。ADV\_BCtool

では境界条件と物性値の付与を行い、ADV\_Meshで部分領域に分割する。そして、ADV\_Solidで先に分割した領域ごとにCPUを割り当てMPIにて並列解析を行う。

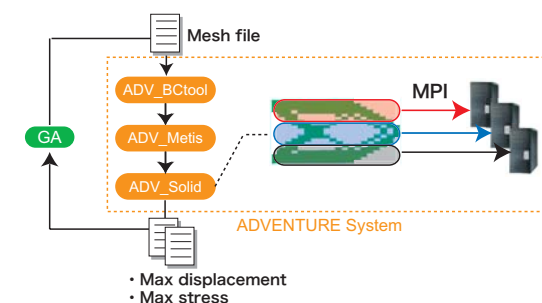


Fig. 6 Parallel analysis using the ADVENTURE System

以下 Fig. 7 は要素数  $N$  が  $N = 400$  と  $N = 8000$  の時の、CPU数の違いによる解析時間の比較である。Fig. 7(a) は  $N = 400$  の位相構造の解析結果を示しており、4CPUが最も短い解析時間となっているが、8CPUでは単一CPUよりも解析時間が長くなる。Fig. 7(b) では、 $N = 8000$  の位相構造の解析結果を示しており、CPU数が増加するにつれ、スケールアップする結果となった。より大規模な位相構造になるにつれ、この傾向は顕著になると考えるため、並列解析が行えるADVENTUREは有効であると考えられる。

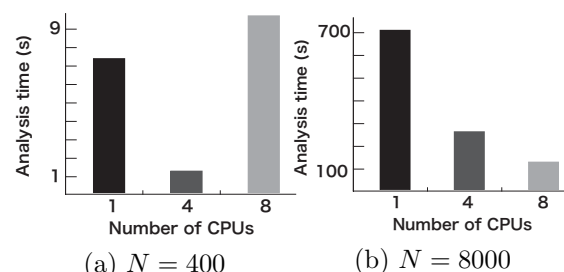


Fig. 7 Comparison of analysis time

### 3.3 Ninf-Gを用いたGAによる 位相構造最適化システム

#### 3.3.1 システム設計

3.1節、3.2節をふまえ、GridRPCシステムNinf-Gを用いたGAによる位相構造最適化を行うシステムを設計する。システム概要を以下 Fig. 8 に示す。

まず、クライアント・サイトでは2章で述べたGAによる位相構造最適化を行う。そして、個体の構造

解析と実行可能領域の制約条件を外れた個体に対して、引き戻し法を適用した後、遠隔地のPCクラスタシステムであるサーバ・サイトの遠隔手続きをGridRPC Ninf-Gを用いて行う。サーバ・サイトのゲートウェイではGlobus ToolkitのGRAMが起動しており、PCクラスタ内部のジョブスケジューラを介して個体情報を内部ノードに分配する。その際、3.2節で述べたように、ADVENTUREシステムを用いてMPIマスタとMPIワーカ間でMPIを利用して並列で構造解析を行う。PCクラスタ間をNinf-Gで通信し、PCクラスタ内でADVENTUREシステムのMPIで通信し、構造解析の並列処理を行う。なお、Grid環境のような計算資源がヘテロな環境では、各サイトの保有するPCクラスタの性能が異なり、構造解析に要する計算時間が異なるが、現段階では計算資源を適応的にスケジューリングする部分に関しては検討を行っておらず、1個体を解析するMPIワーカの数に静的に決定するものとしている。

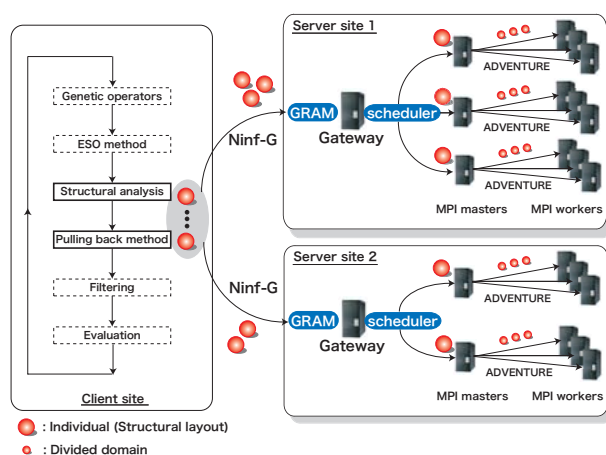


Fig. 8 Overview of the proposed system

### 3.3.2 システム実装

前述したシステムのPCクラスタ間の通信は、GridRPC Ninf-GのJava実装モデルである、Ninf-G Javaを用いて実装を行う。またサーバ・サイトのゲートウェイにはGlobus Toolkit(バージョン4.0.1)を用い、ジョブスケジューラにはOpen PBS(Torqueバージョン1.2)を用いる。

## 4 まとめと今後の課題

構造形態の最適化を行うためには、構造物の形状のみならず位相を最適化する必要がある。本研究では、位相構造最適化にGAを用いることを検討している。GAを位相構造最適化に適用することで、従来の手法では創生できない、複数の制約条件を考慮した、より合理的な構造形態を創生できることを示した。そして、GAを適用する上で問題となってくる、多くの計算資源の確保にGridRPCシステムNinf-Gを利用することを検討した。PCクラスタ間を代表的なGridミドルウェアであるNinf-Gで通信し、PCクラスタ内ではADVENTUREシステムの並列構造解析を利用することで、構造解析も並列実行することを検討した。

今後の課題としては、Ninf-G Javaを用いたクライアント・サイトとサーバ・サイトの実装を行い、複数のPCクラスタを用いた評価実験を行う。

## 参考文献

- 1) Bendsøe, M.P. and Kikuchi, N., *Generating optimal topologies in structural design using a homogenization method*, Computer Methods in Applied Mechanics and Engineering, Vol.71, pp.197-224, 1988.
- 2) Bendsøe, M. P., *Optimal shape design as a material distribution problem*, Struct. Optimiz., Vol.1, pp.193-202, 1989.
- 3) Xie, Y.M. and Steven, G.P. *Evolutionary Structural Optimization*, Springer-Verlag, 1997.
- 4) H. Saito, E. Kita, and Y. M. Xie, *Three-dimensional Structural Optimization Using Cellular Automata*, Advances in Applied Mechanics (Proc ACAM2005), pp.781-788, 2005.
- 5) D. E. Goldberg, *Genetic Algorithms in search, Optimization and Machine Learning* Addison-Wasley Publishing Company, 1989.
- 6) Foster, I., Kesselman, C. and Tuecke, S., *The Anatomy of the Grid: Enabling Scalable Virtual Organizations*, International Journal of Supercomputer Applications, 2001.
- 7) 小野功, 水口尚亮, 中島直敏, 小野典彦, 中田秀基, 松岡聡, 関口智嗣, 楯真一, *Ninf-1/Ninf-G*を用いたNMR蛋白質立体構造決定のための遺伝アルゴリズムのグリッド化, 先進的計算基盤システムシンポジウム SACSIS 2005, IPSJ Symposium Series Vol. 2005, No. 5, pp.143-151, 2005.
- 8) 田中良夫, 中田秀基, 平野基孝, 佐藤三久, 関口智嗣, *Globus*によるGridRPCシステムの実装と評価, 情報処理学会ハイパフォーマンスコンピューティング研究会, Vol.2001, No.77, pp.165-170, 2001.
- 9) ADVENTURE Project : <http://adventure.q.t.u-tokyo.ac.jp/jp/>