

深層学習により反復解法の 計算時間推定を行うための基礎的検討

東京大学大学院 新領域創成科学研究科
森田直樹

2018/2/5

目次

- 目次
- 概要
- 背景
- 目的
- 深層学習による反復解法の計算時間の推定
- まとめ

- 試行錯誤的に決定している
線形ソルバの条件設定コストを削減する
- 反復法の反復回数を機械学習で推定する
- 係数行列から得る特徴量を用いて学習し反復回数を推定する

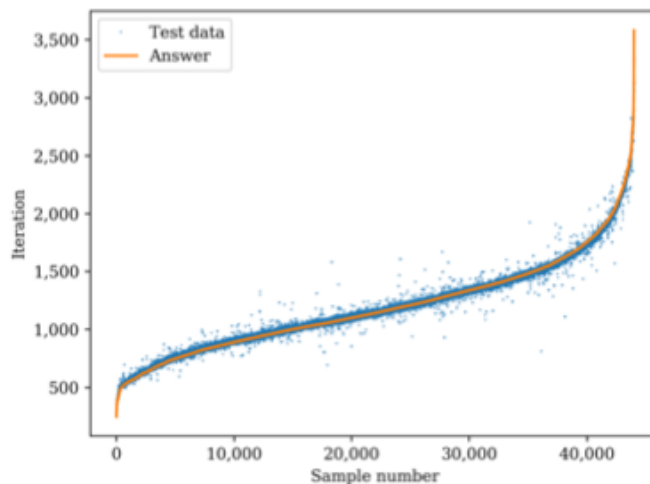


Fig. 1 機械学習による反復回数の推定例

- **解きたい問題**に対しどの線形ソルバを利用するか？
- どのような行列か
 - ・ 非零要素の値・配置が対称 / 非対称
 - ・ 非零要素が実数 / 複素数
 - ・ 正定値対称性がある / ない
 - ・ 行列の条件数
- どのような解析手法か
 - ・ 静的解析 / 動的解析 / 固有値解析
 - ・ 線形解析 / 非線形解析
- どのような並列計算環境か
 - ・ 問題の自由度 / 分割領域の自由度
 - ・ 使用可能なCPUコア数 / 領域分割数
 - ・ MPIフラット並列 / OpenMPスレッド並列
 - ・ 計算機の単体性能・インターコネクト性能

- 解きたい問題に対しどの**線形ソルバ**を利用するか？
- 直接法
 - Multifrontal法
 - Nested Dissection法
 - SPIKE法
- 反復法
 - CG法
 - BiCGStab法
 - GMRES法
- 反復法前処理
 - 対角スケーリング前処理
 - 対称逐次過緩和前処理
 - ILU (p) 前処理
 - マルチグリッド前処理

- いかなる条件においても
効率的に計算できる線形ソルバは存在しない
 - 並列性能の観点からは一般に反復法が選択される
 - 線形動的解析では直接法で一度行列分解を行えば
その後は前進・後退代入のみでよい
- 選定時の大半が利用者個人の経験に依存し
時間コスト・計算資源コストを無視できない

目的

- 問題・実行環境にあわせた最適な線形ソルバの条件を自動選定できる手法の開発

深層学習による 反復解法の計算時間の推定

有限要素解析と線形ソルバ

- 線形ソルバの機能：連立一次方程式の解を求める
 - 得られた解が要求精度を満たす

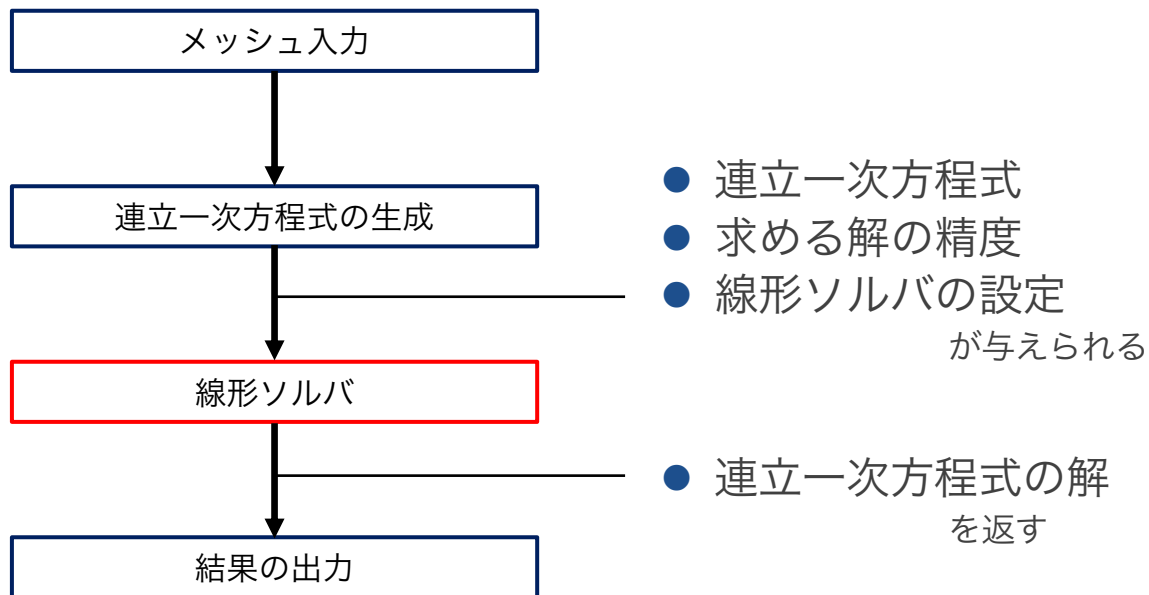


Fig. 2 有限要素解析の概略図

理想の線形ソルバはどのようなものか？

- 解析内容・計算機にあわせて
最適な線形ソルバの条件を決定してくれる



- 解が要求精度を満たす、最も計算時間が短くなる条件

- 最も計算時間が短くなる条件を深層学習を用いて推定する
 - DNN (Deep Neural Network) を利用する

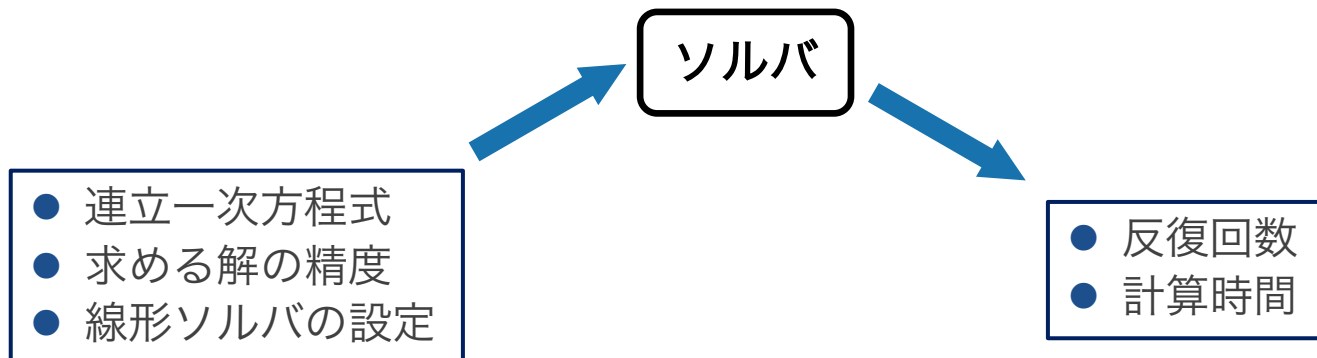


Fig. 3 機械学習による反復回数推定の概略図

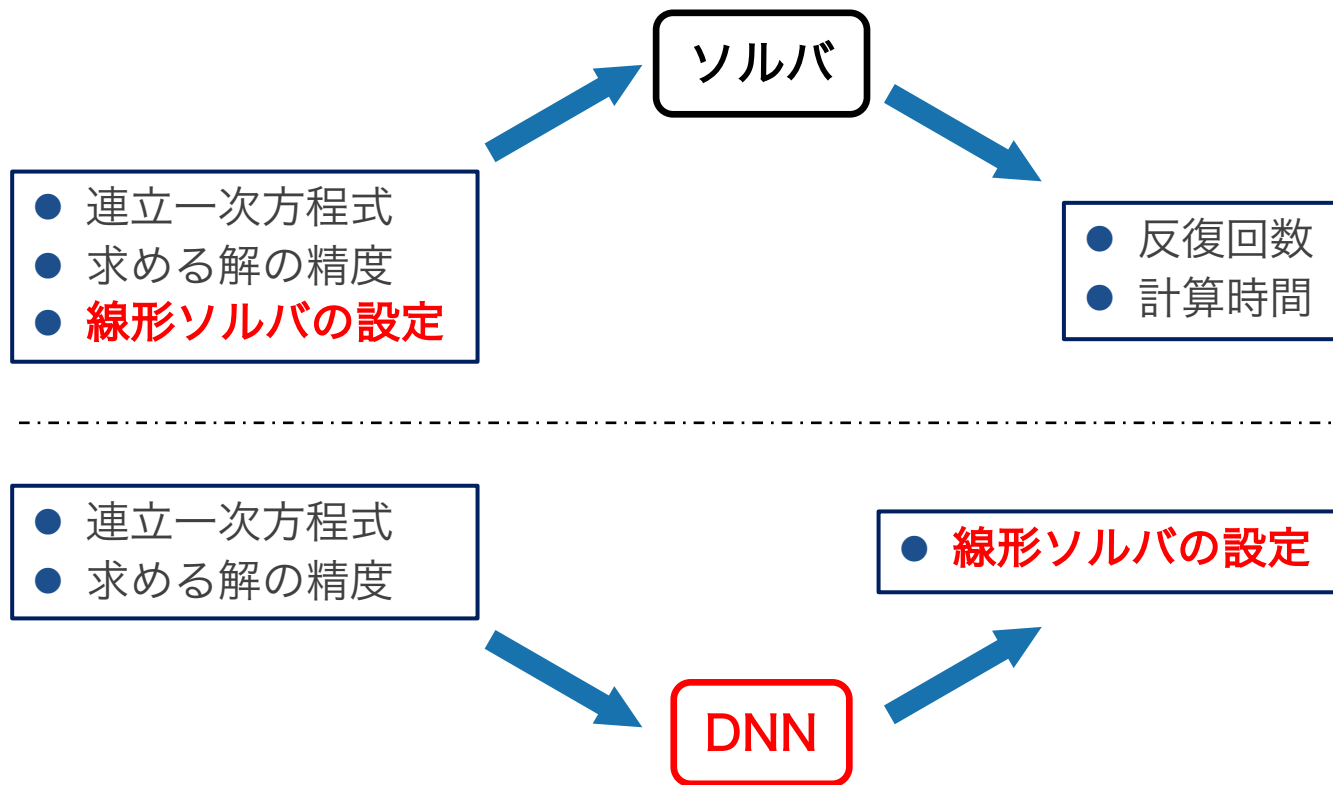


Fig. 4 機械学習による反復回数推定の概略図

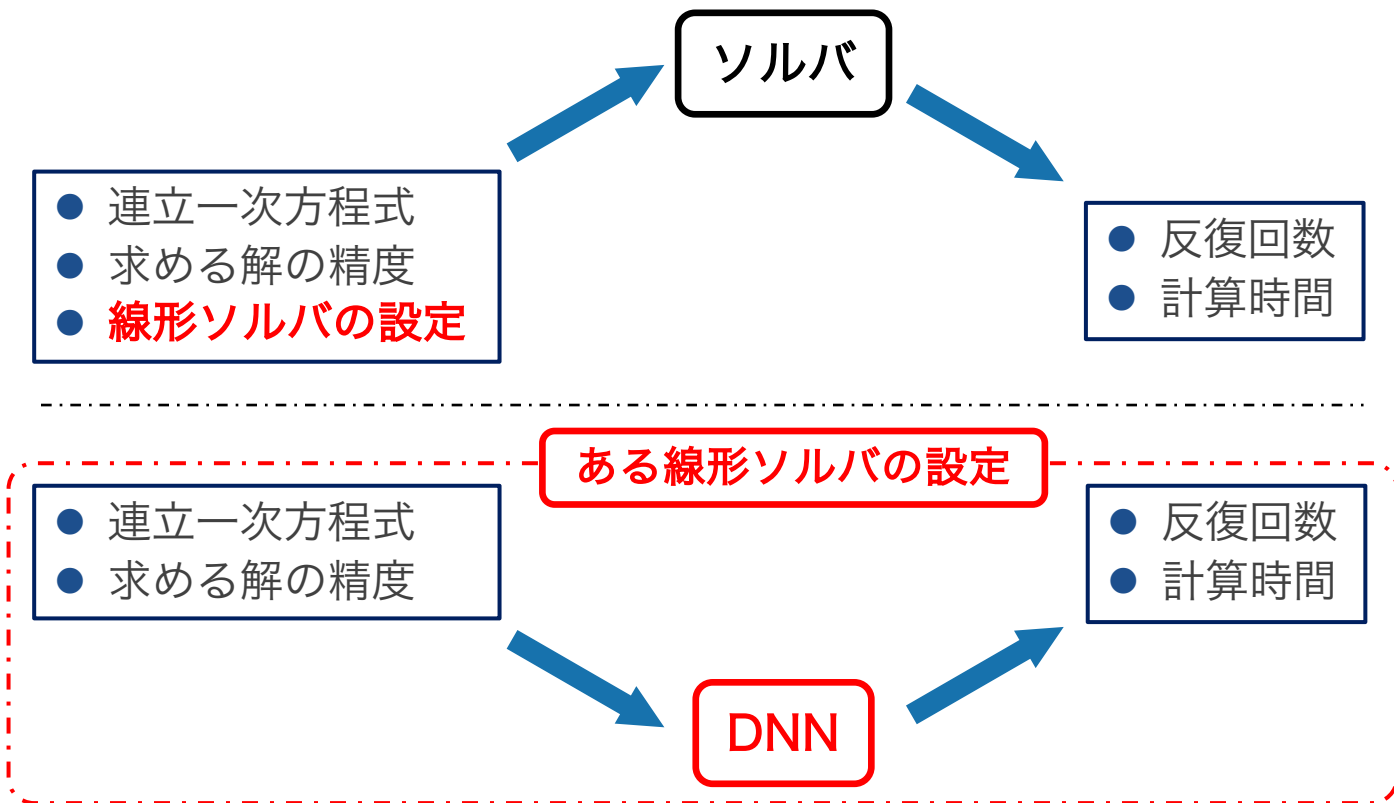


Fig. 5 機械学習による反復回数推定の概略図

深層学習による反復回数推定の概略図 [1/3]

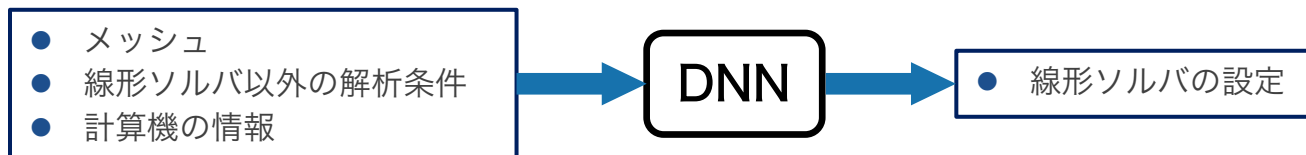


Fig. 6 機械学習による反復回数推定の概略図

深層学習による反復回数推定の概略図 [2/3]

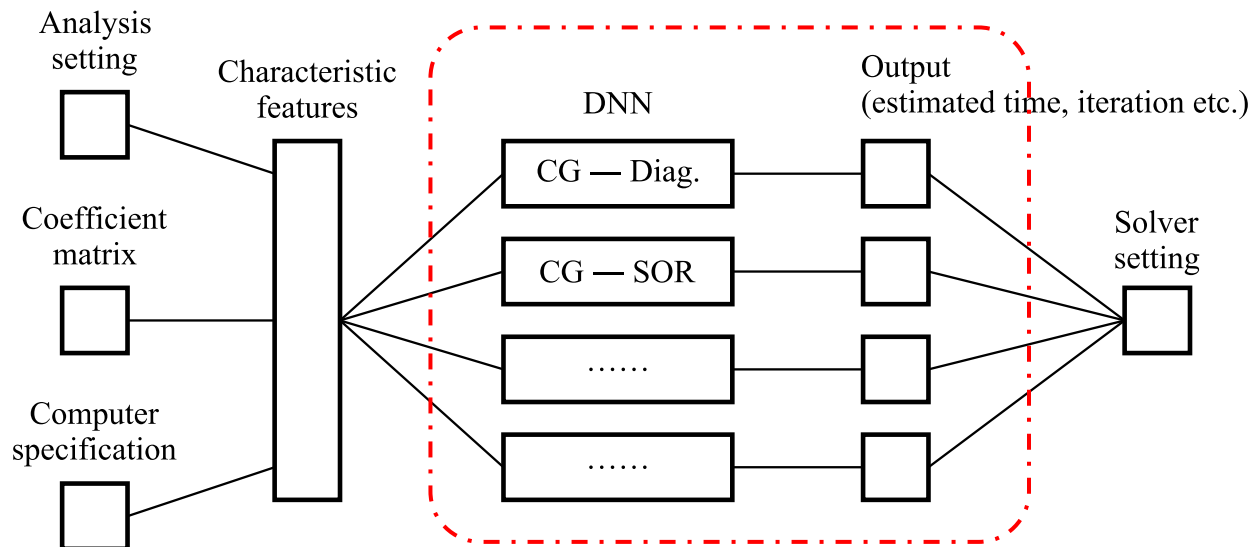


Fig. 7 機械学習による反復回数推定の概略図

本研究で考慮した制約条件

- 本研究では以下の制約のもと
機械学習により線形ソルバの選定が可能か検討する
- 線形ソルバ
 - 共役勾配法—前処理なし
 - 共役勾配法—対角スケーリング前処理
 - 共役勾配法—対称逐次化緩和前処理
- 逐次計算
- 計算機環境を考慮しない
- 収束判定 1.0×10^{-8}

本研究で考慮した制約条件による概略図

- 線形ソルバを自動選定するためには
精度良く反復回数を推定できればよい

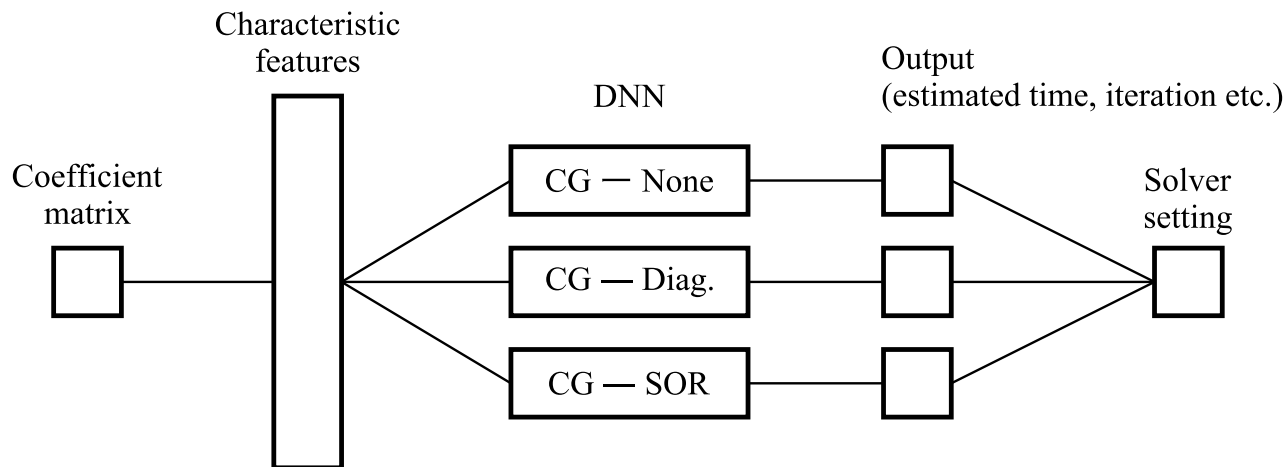


Fig. 8 機械学習による反復回数推定の概略図

反復回数の推定における深層学習の利点

- 学習データの準備が最も重要
 - 実社会からの実測データを必要としない
 - 自動的・大量に学習データを生成できる

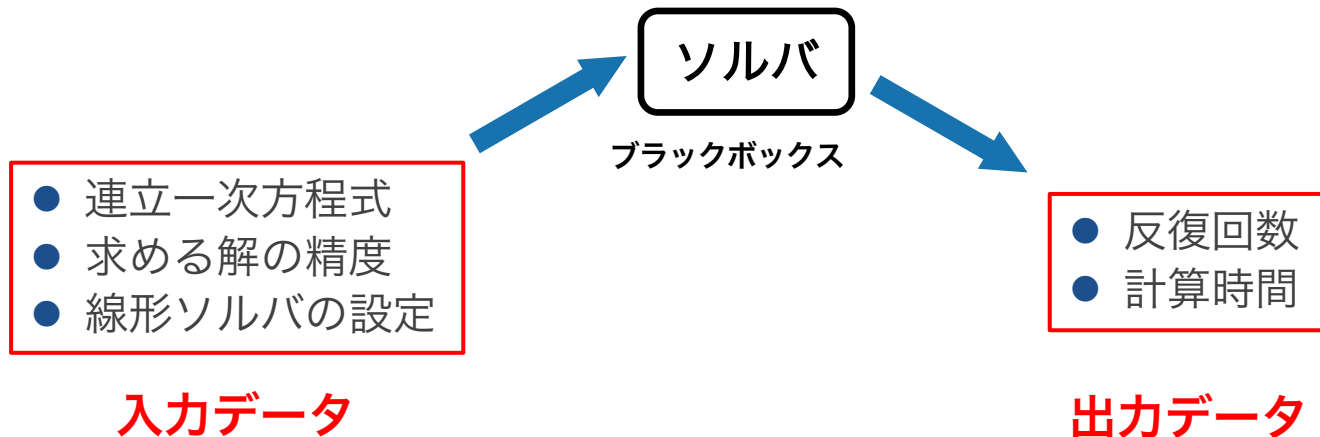


Fig. 8 機械学習による反復回数推定の利点

学習データの自動生成 [1/3]

- 様々な条件（メッシュ・線形ソルバの設定）で実際に計算し、学習データを生成すればよい
- 自動メッシャー用いてデータを生成するが組み合わせが爆発してしまう
- メッシュは、六面体一次要素のみから構成することした

- メッシュは以下の自由度から自動生成することとした

- 伸び $\varepsilon_x, \varepsilon_y, \varepsilon_z$
- 曲げ $M_{yx}, M_{zx}, M_{xy}, M_{zy}, M_{xz}, M_{yz}$
- せん断 $\tau_{xy}, \tau_{yz}, \tau_{zx}$
- ねじり ϕ_x, ϕ_y, ϕ_z
- 辺の分割数 p_x, p_y, p_z
- 要素のヤング率 E , ポアソン比 ν

● 具体的には以下のように自由度を組み合わせる

- 伸び $\varepsilon_x, \varepsilon_y, \varepsilon_z$ をそれぞれ [考慮する, 考慮しない].
- 曲げ $M_{yx}, M_{zx}, M_{xy}, M_{zy}, M_{xz}, M_{yz}$ のうち, M_{yx}, M_{xy}, M_{xz} をそれぞれ [考慮する, 考慮しない].
- せん断 $\tau_{xy}, \tau_{yz}, \tau_{zx}$ をそれぞれ [考慮する, 考慮しない].
- ねじり ϕ_x, ϕ_y, ϕ_z は考慮しない.
- 辺の分割数 p_x, p_y, p_z は, [32, 64, 128] を表 3.1 のように組み合わせる値をとる.
- 回転 $\theta_x, \theta_y, \theta_z$ がそれぞれ [0, 45] [deg.] の値をとる.
- 要素のヤング率 E は [20,000, 200,000] の値をとる.
- ポアソン比 ν は [0.2, 0.4] の値をとる.

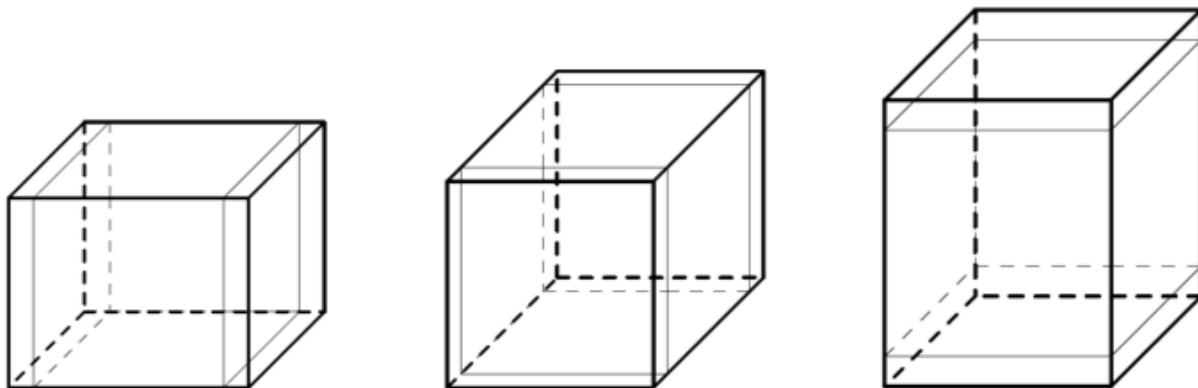


Fig. 3.2 Tensile modes of a cuboid model ($\epsilon_x, \epsilon_y, \epsilon_z$)

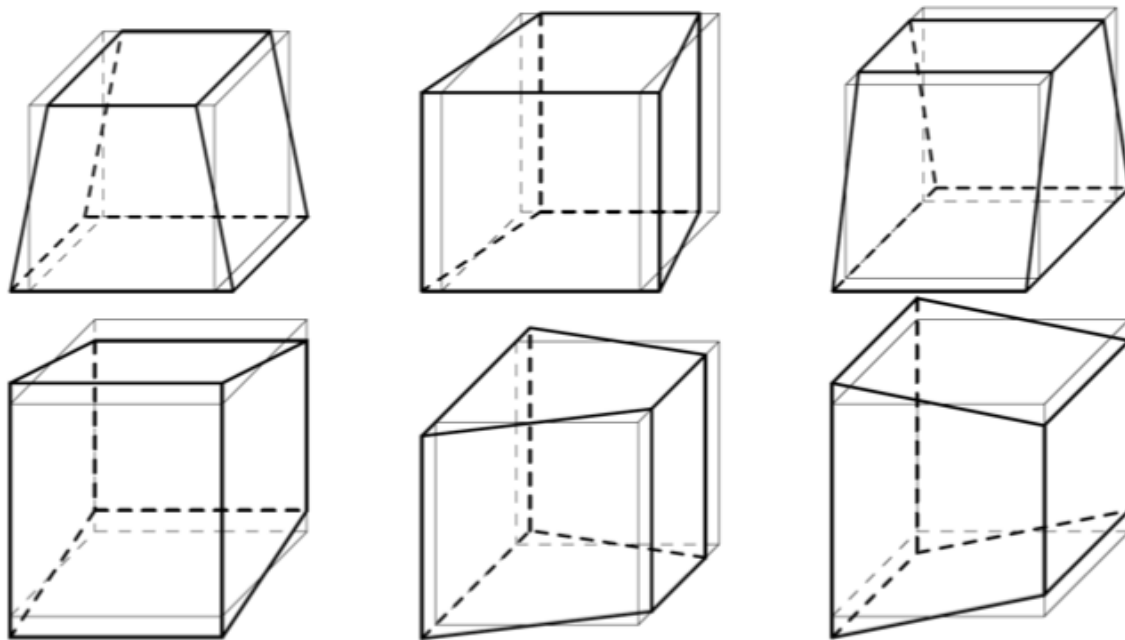


Fig. 3.3 Bending modes of a cuboid model (M_{yx} , M_{zx} , M_{xy} , M_{zy} , M_{xz} , M_{yz})

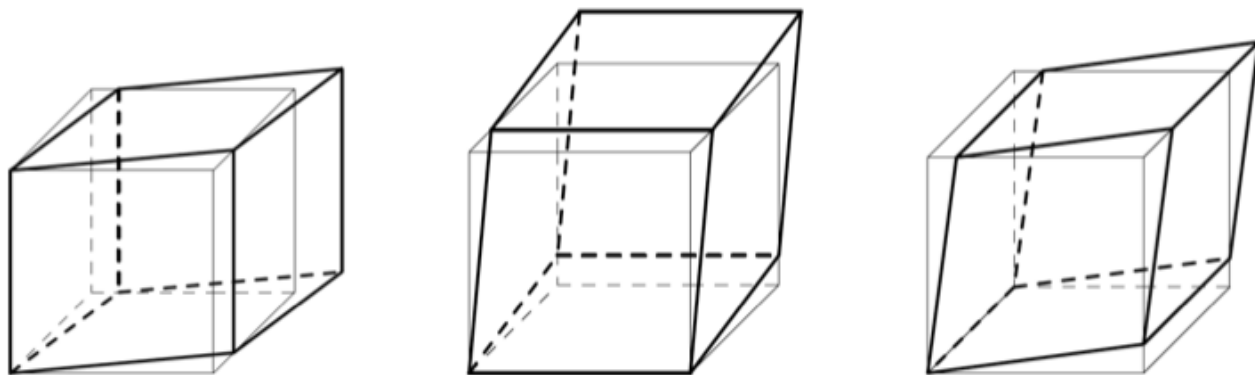


Fig. 3.4 Shear modes of a cuboid model (τ_{xy} , τ_{yz} , τ_{zx})

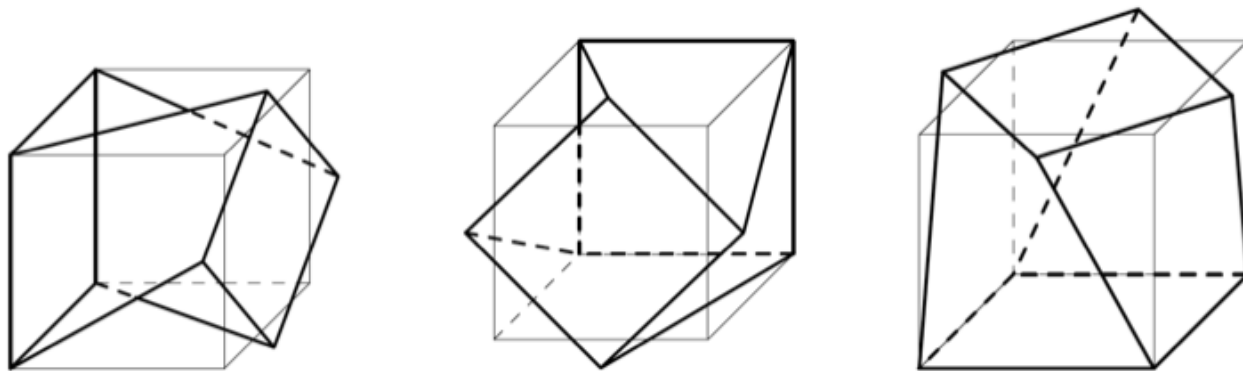


Fig. 3.5 Torsion modes of a cuboid model (ϕ_x, ϕ_y, ϕ_z)

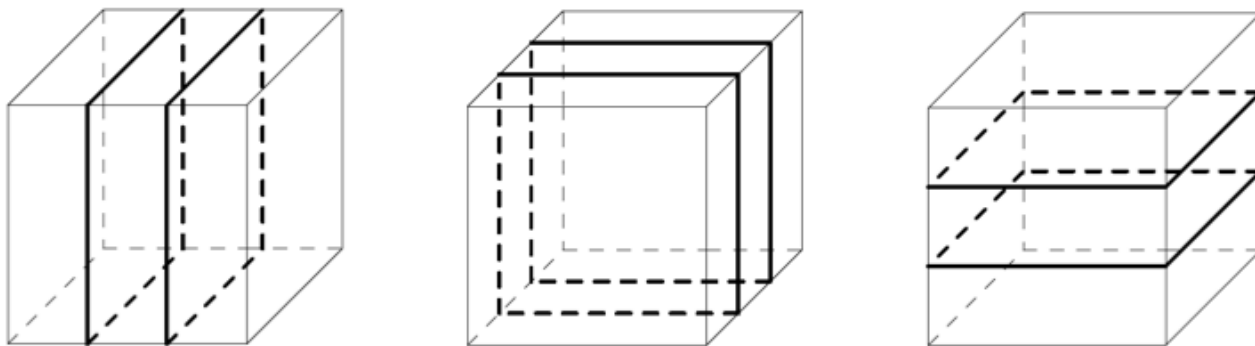


Fig. 3.7 Number of division of a cuboid model (p_x, p_y, p_z)

Table 3.1 Combination of number of division of a cuboid model

ID	p_x	p_y	p_z	num. of node	D.O.F
1	32	32	32	35,937	107,811
2	32	32	64	70,785	212,355
3	32	64	32	70,785	212,355
4	32	64	64	139,425	418,275
5	64	32	32	70,785	212,355
6	64	32	64	139,425	418,275
7	64	64	32	139,425	418,275
8	64	64	64	274,625	823,875
9	64	64	128	545,025	1,635,075
10	64	128	64	545,025	1,635,075
11	64	128	128	1,081,665	3,244,995
12	128	64	64	545,025	1,635,075
13	128	64	128	1,081,665	3,244,995

メッシュ生成例

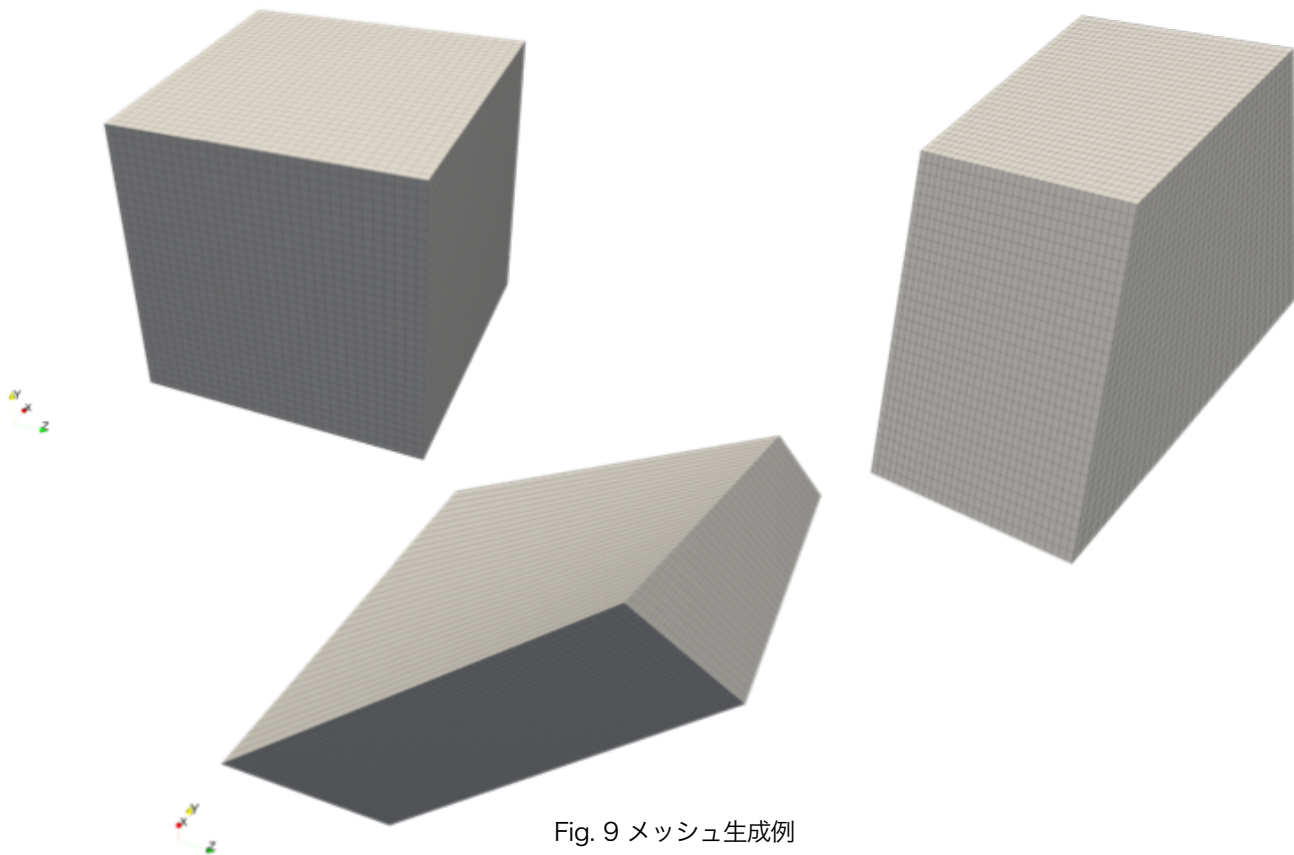


Fig. 9 メッシュ生成例

データ生成数

- 生成メッシュ数：212,992
- データは以下の線形ソルバの条件で計算し、反復回数を得た
 - 共役勾配法—前処理なし
 - 共役勾配法—対角スケーリング前処理
 - 共役勾配法—対称逐次過緩和前処理
- 自重解析を実施し、右辺ベクトルの影響は考慮しない
- 計算にはFrontISTRを用いた

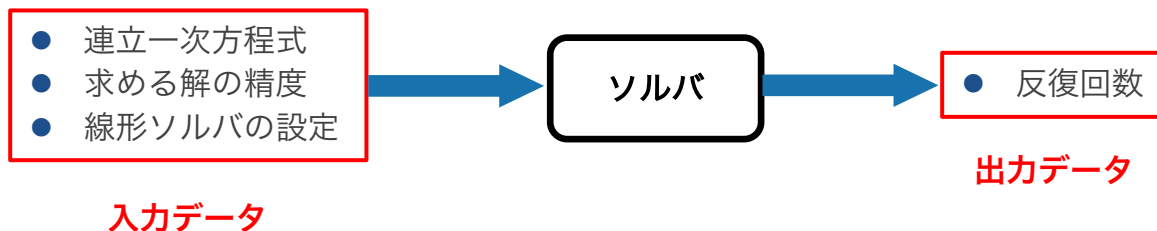


Fig. 10 入力データと出力データの関係