東京大学本郷キャンパス 工学部8号館 85講義室 (地下1階)

関連ソフトウェア：FrontISTRベース高度溶接シミュレータのプリポスト

2018年7月9日

第44回FrontISTR研究会

＜FrontISTR関連プロジェクトの研究成果紹介＞

ポスト「京」重点課題⑧サブ課題Eのホームページ

文部科学省 フラッグシップ2020プロジェクト
ポスト「京」重点課題⑧「近未来型ものづくりを先導する革新的設計・製造プロセスの開発」
サブ課題E「新材料に対応した高度成形・溶接シミュレータの研究開発」

プロジェクト概要

高度溶接シミュレーション技術を開発し、溶接工程における溶接順序探索および逆ひずみ量推定の高精度化・高速化を行うことが目的である。平成29年度までの達成目標として、入熱による熱弾塑性解析の計算精度を検証し、数mmのオーダーの溶融条件を考慮した大規模並列計算性能を検証する。平成31年度までの達成目標として、ターゲット問題における部品規模の溶接解析の計算精度を従来アプリと比較し、開発するアプリの優位性を示す。そして、全体規模の溶接解析結果を実験値と比較し、開発するアプリの予測精度を検証する。

・プロジェクトメンバー

奥田洋司 教授 (担当責任者)
橋本学 講師 (実施担当)
林雅江 特任研究員 (実施担当)
井原遊 RA (実施担当)
生野達大 RA (実施担当)

謝辞

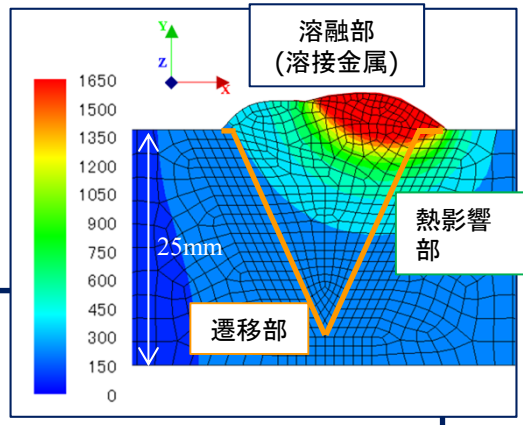
本研究開発は，文部科学省ポスト「京」重点課題⑧
「近未来型ものづくりを先導する革新的設計・製造プロセスの開発」の一環として実施したものです

FrontISTRベース高度溶接シミュレータのプリポスト

1. プリポストの概要
2. ユーザ向けの紹介
3. ディベロッパー向けの紹介

サブ課題E　新材料に対応した高度成形・溶接シミュレータの研究開発

- 溶接による収縮を高精度に予測するためには、母材の接触状態を詳細に計算する必要がある
- 数m規模の解析領域に存在する数mmの溶融部を数μmで解像することが可能なソルバーを開発する
  数千億～数兆要素 (メッシュの粗密あり、溶融部に詳細メッシュ)



超大規模・高精度連成解析ソルバー

✓ アセンブリ／接触問題の大規模解析が可能な並列反復法
(従来の溶接シミュレータは直接法ベースであったが，
直接法では解けない，十分な並列性能を発揮できない規模を対象とする)
- 「京」: 1日で十億要素の静解析が限界である
- ポスト「京」: 大型機械フレーム全体 (自動車や重機械など) に存在する溶融部での接触状態を詳細に解析可能にする

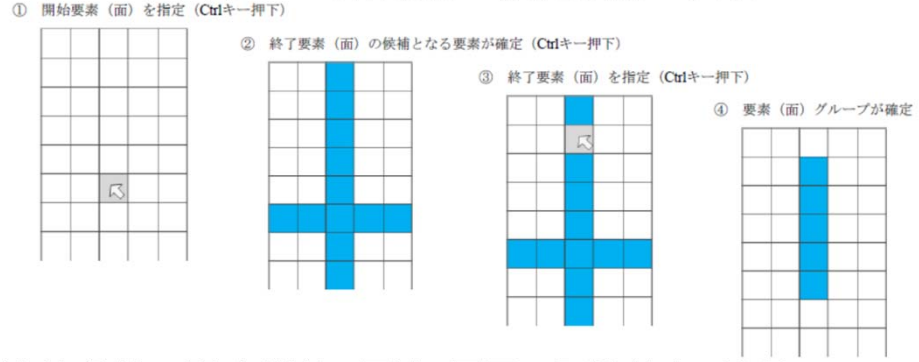
✓ 大規模双方向連成解析 (熱伝導・弾塑性変形)
(材料非線形性，幾何学的非線形性，接触非線形性が大きい問題である)
- 「京」: 1日で数億要素規模の片方向連成解析でも不可能である
- ポスト「京」: 支配方程式系を一括して，双方向連成解析を可能にする

✓ プレス成形時のスプリングバックの影響を考慮した溶接解析
(他のソフトウェアでは，プレス成形時のスプリングバックを考慮していない)
- 従来解析: プレス成形時のスプリングバックによる残留応力を無視
- 本解析: プレス成形／溶接の一連の工程を解析可能なプリポストによって，プレス成形のスプリングバックによる残留応力を溶接の初期条件に考慮する

プリポストの研究開発 (1/2)

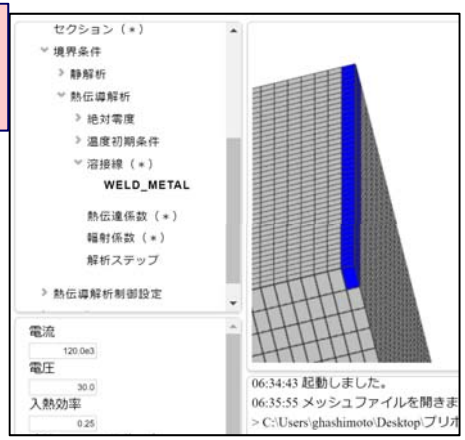
達成済みの成果

- 開発環境Electronを使用してウェブブラウザで動作するプリポストの開発
  - Windows、Mac、Linuxなどの様々なOSでプリポストが動作
  - Electronでは、プログラマブルなブラウザが提供されるため、ブラウザ上で動作するデスクトップアプリをHTML、JavaScript、CSSで比較的容易に開発可能
- 数千万自由度の問題に対して動作確認済み
- 溶接解析の機能（溶接線，溶接条件の設定機能など）を拡張し、動作検証を実施
  - マウスのクリックによる溶接線の設定機能の実装

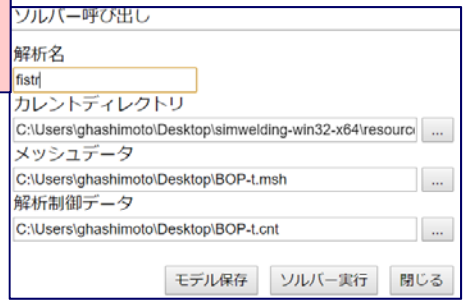


  - 溶接条件の設定機能 (電流・電圧，入熱効率，溶接トーチの移動速度，溶接源の幅，溶接開始時間の入力)の実装
  - ソルバー呼び出し機能の実装

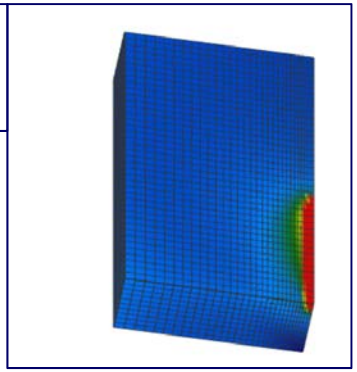
溶接条件の設定



ソルバー呼び出し



溶接解析の結果表示



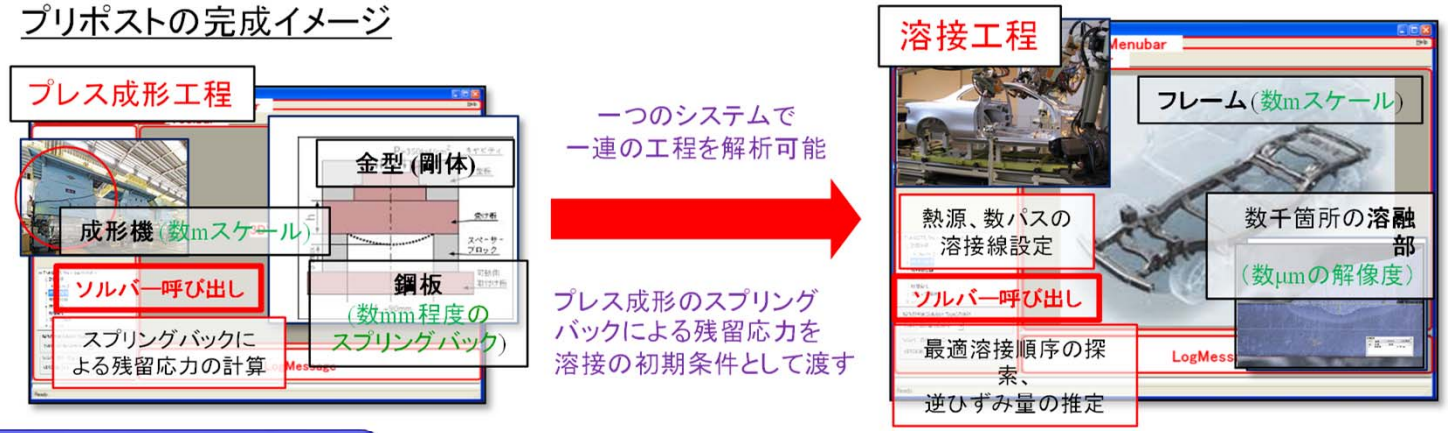
プリポストのプロトタイププログラムの完成

プリポストの研究開発 (2/2)

プリポストのプロトタイププログラムの特徴
- 要素タイプの追加が容易
- イベント (解析制御ファイル設定) の追加が容易
- 多言語へ対応が容易

※ 作業手順をマニュアルとして整備済み

プリポストの完成イメージ



平成30年度達成目標

- 溶接解析の機能 (複数パスの設定機能など) を拡張し、動作検証の実施
- プレス成形解析機能 (補助的なツール) からのデータ渡しについて検討

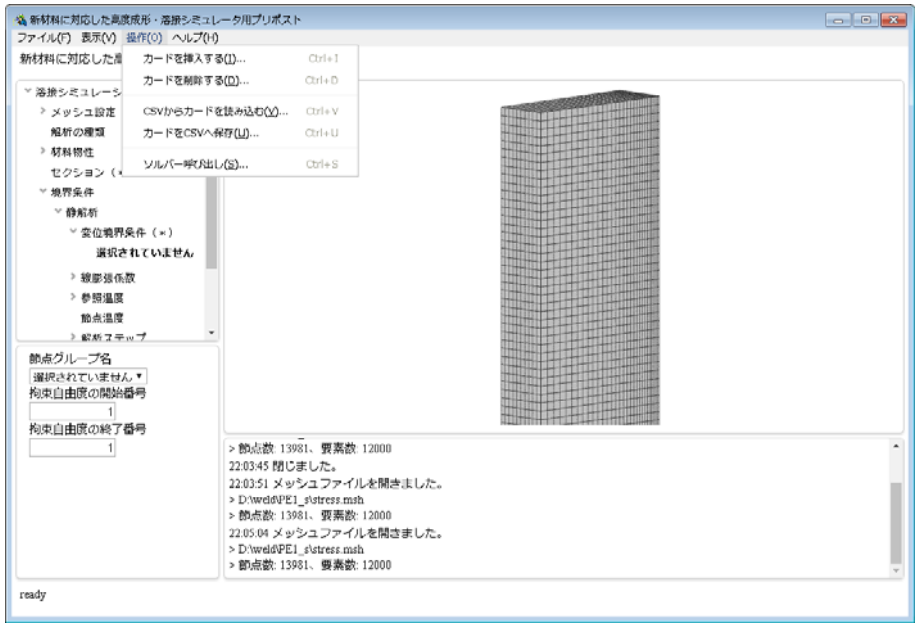
最終達成目標

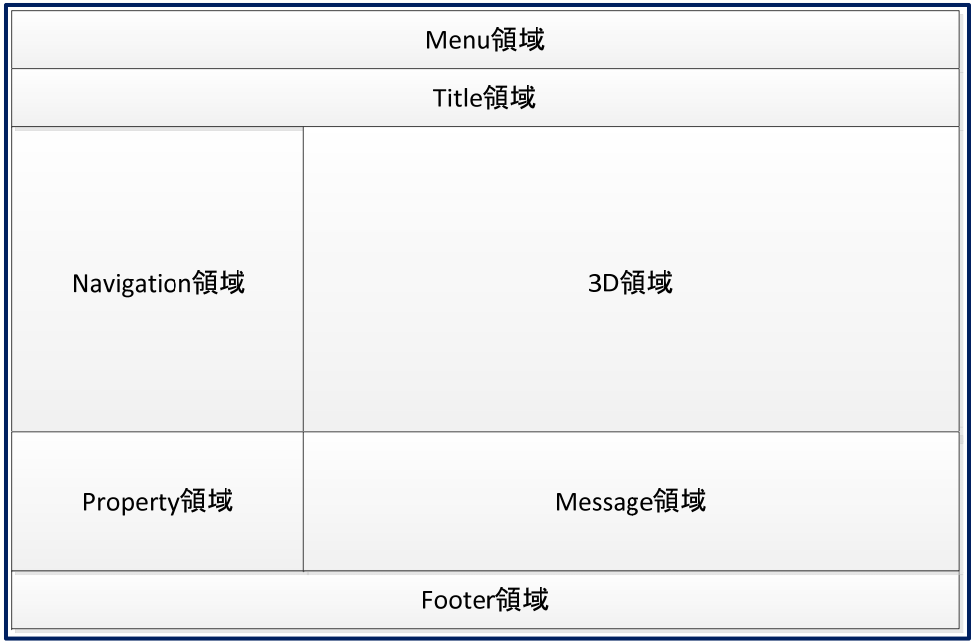
- プレス成形のスプリングバックによる残留応力を溶接の初期条件として渡す機能を実装し、動作検証の実施

FrontISTRベース高度溶接シミュレータのプリポスト

1. プリポストの概要
2. ユーザ向けの紹介
3. ディベロッパー向けの紹介

画面構成

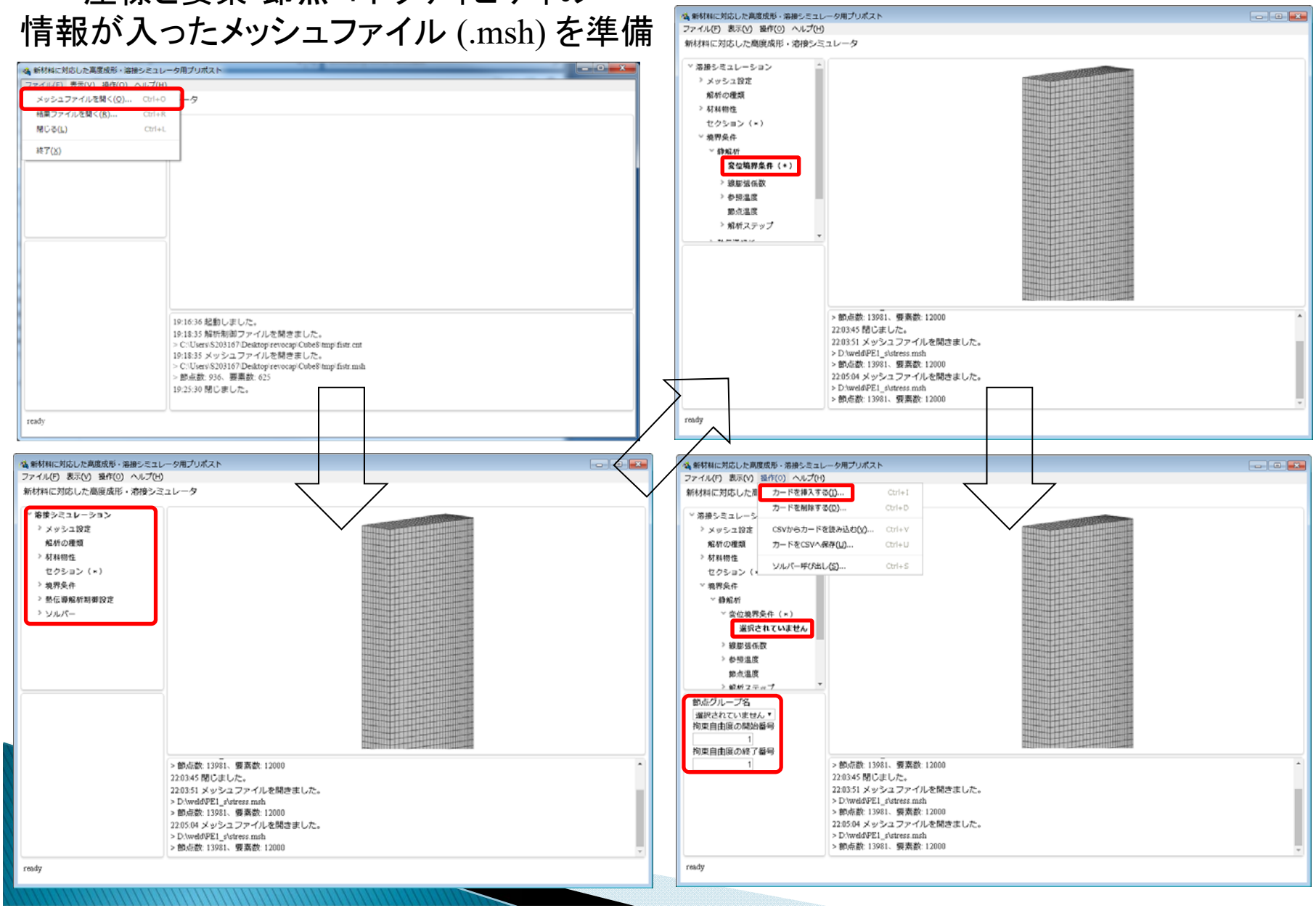




No.	領域名	説明
1	Navigation領域	３D領域に表示されるモデルの構造を表示する。 構造的操作の対象となるオブジェクト等の選定を支援する。
2	Property領域	３D領域に表示されるモデルの属性を表示する。 属性の状態やその変更操作を支援する。
3	3D領域	モデルを３D表示する。
4	Message領域	各種操作等の履歴を表示する。
5	Menu領域	コマンドのメニューを表示する。
6	Title領域	タイトルを表示する。
7	Footer領域	システムの状態を表示する。（Ready）

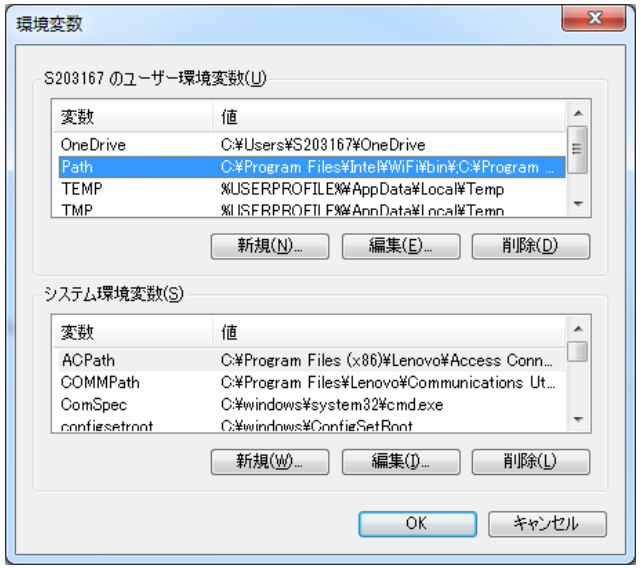
操作手順 (プリ)

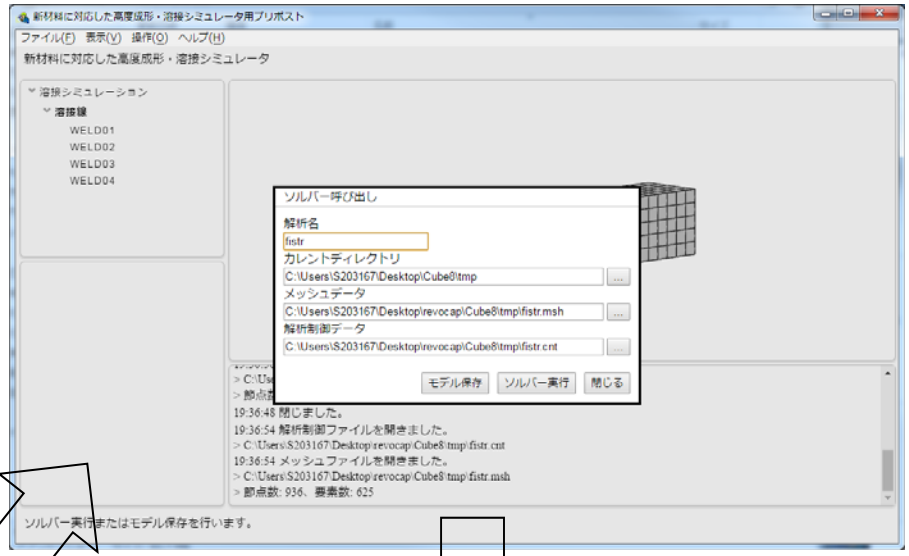
座標と要素-節点コネクティビティの
情報が入ったメッシュファイル (.msh) を準備

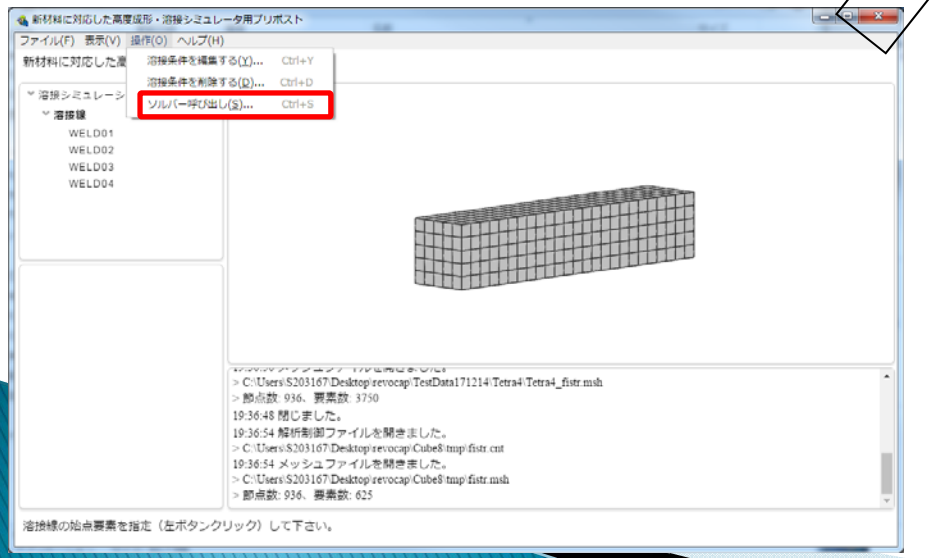


操作手順 (ソルバー)

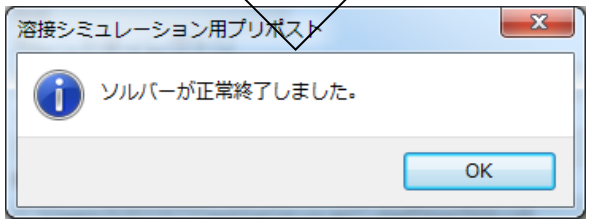
FrontISTR実行ファイル (fistr1) が
置かれたフォルダのパスを設定





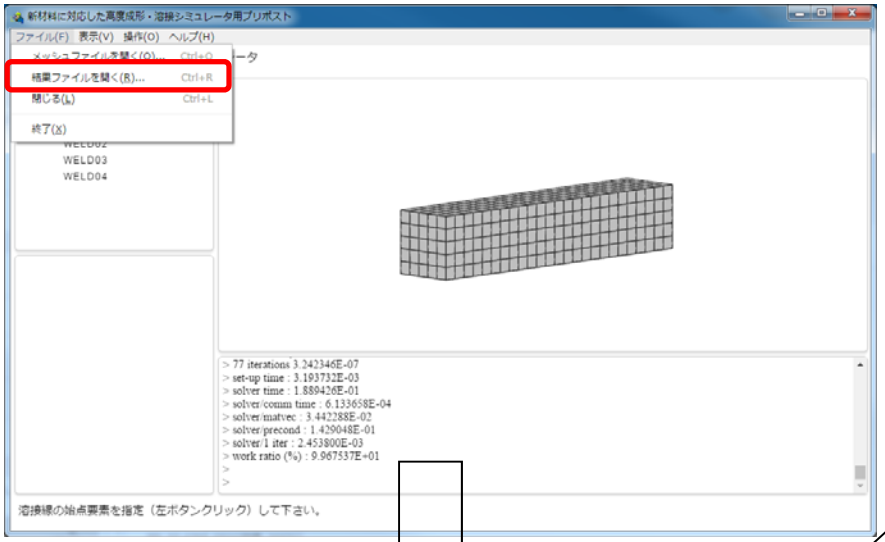


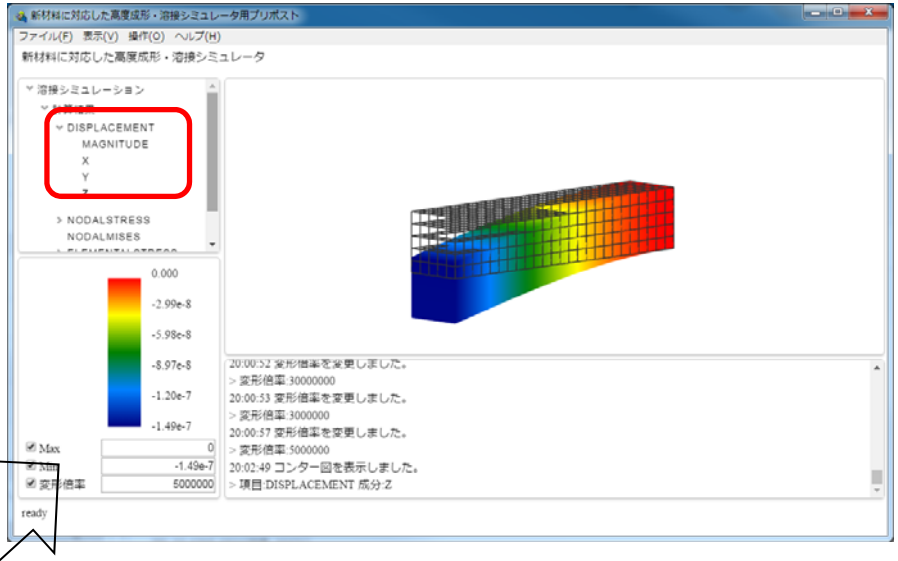
メッシュファイル，解析制御ファイル，
全体制御ファイルを生成し，
FrontISTRプログラムを実行

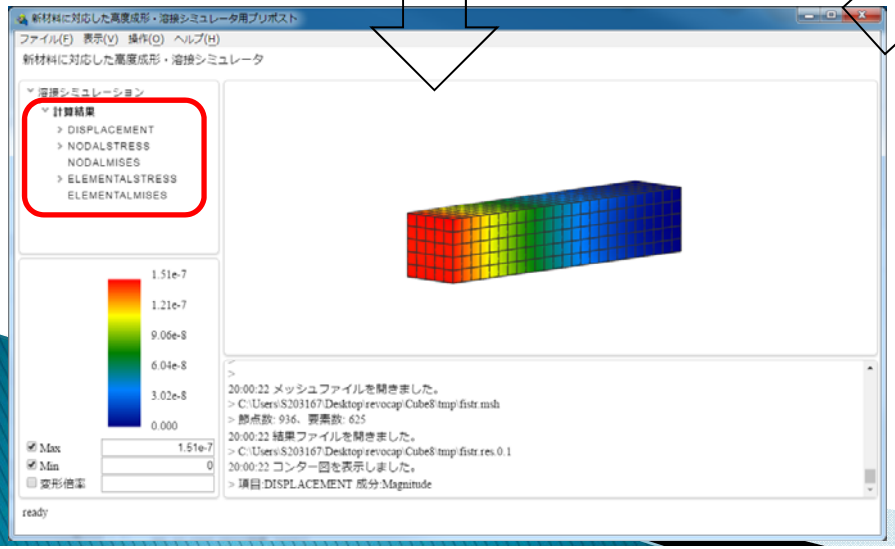


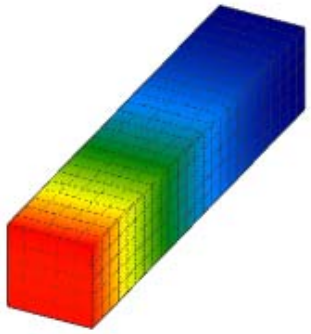
操作手順 (ポスト)

使用したメッシュファイル (.msh) と
計算結果ファイル (.res) を準備

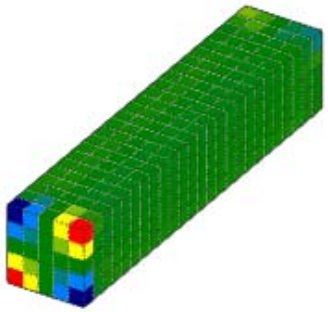








nodeの場合
(グラデーション)



elementの場合
(クライテリア)

デモ

試験片の溶接シミュレーションのデモ

1. 熱伝導解析
2. 弾塑性解析

プリポストに読み込むファイル

読み込むファイル

No.	操作		ファイル種類	説明
	メニュー	Item		
1	ファイル	メッシュファイルを開く	メッシュデータ	
2	ファイル	結果ファイルを開く	メッシュデータ 結果データ	メッシュ→結果の順でファイルを指定する。

メッシュ読み込み機能にて対応する要素タイプ

No.	要素種類	要素番号	対応Version	説明
1	ソリッド要素	341	1.1.X以降	4節点四面体一次要素
2	ソリッド要素	342	1.1.X以降	10節点四面体二次要素
3	ソリッド要素	361	1.0.X以降	8節点六面体一次要素
4	ソリッド要素	362	1.1.X以降	20節点六面体二次要素

※ ソルバーの呼び出しでは，メッシュファイル，解析制御ファイル，全体制御ファイルを読み込む

FrontISTRの入出力ファイルの対応状況 (1/2)

No.	ファイル種類	拡張子	対応Version	対応レコード
1	解析制御データ	cnt	1.1.X以降	溶接線 ✓ !WELD_LINE 熱伝導解析用制御データ ✓ !SFILM ✓ !SRADIATE ✓ !STEP ✓ !HEAT 変位境界条件 ✓ !BOUNDARY 静解析用制御データ ✓ !MATERIAL ✓ !EXPANSION_COEFF ✓ !REFTEMP ✓ !TEMPERATURE ✓ !STEP
2	全体制御データ hecmw_ctrl.dat	dat	1.1.X以降	メッシュファイルのリンク ✓ !MESH, NAME=fstrMSH ✓ !MESH, NAME=mesh 解析制御ファイルのリンク ✓ !CONTROL,NAME=fstrCNT その他のリンク ✓ !RESTART,NAME=restart_out ✓ !RESULT,NAME=fstrRES ✓ !RESULT,NAME=result ✓ !RESULT,NAME=vis_out

FrontISTRの入出力ファイルの対応状況 (2/2)

No.	ファイル種類	拡張子	対応Version	対応レコード
3	メッシュデータ	msh	1.0.X	要素361 ✓ !NODE ✓ !ELEMENT
			1.1.X以降	要素341、342、361、362 ✓ !NODE ✓ !ELEMENT 要素・面・節点グループ ✓ !EGROUP ✓ !SGROUP ✓ !NGROUP 材料 ✓ !MATERIAL ✓ !SECTION 境界条件 ✓ !ZERO ✓ !INITIAL CONDITION
4	結果データ	res(.*(.*))	1.0.X	要素361 DISPLACEMENT（絶対値）
			1.1.X以降	要素341、342、361、362 DISPLACEMENT（変位、絶対値） NodalSTRESS（節点応力） NodalMISES（Mises応力） NodalSTRAIN（節点ひずみ） ElementalSTRESS（要素応力） ElementalMISES （要素Mises応力） ElementalSTRAIN（要素ひずみ）

FrontISTRベース高度溶接シミュレータのプリポスト

1. プリポストの概要
2. ユーザ向けの紹介
3. ディベロッパー向けの紹介

アプリケーションの開発環境「Electron」

- 開発アプリケーションがウェブブラウザ上で動作
- Windows／Mac OS X／Linux上で動作する
  クロスプラットホームアプリケーションを開発可能
- HTML，JavaScript，CSSのWeb技術を用いて開発できる
  フレームワーク

<u>開発環境の構築手順</u>

1. Node.jsのインストール
2. Electronのインストール
3. アプリケーションのデバッグ実行

(アプリケーションのパッケージ作成)

Node.jsのインストール

ウェブサイト（https://nodejs.org/ja/）からNode.jsをダウンロード

Windows OSから前述のウェブサイトにアクセスすると
インストーラがダウンロード可能

Electronのインストール

1. Windows OSの場合，スタートメニューから
   [Node.js]→[Node.js command prompt]を選択し，
   コマンドプロンプトを起動

2. Electronのインストールには次のコマンドを実行

   > npm -g install electron-prebuilt

※ npm (node package manager) : Node.jsのパッケージ管理ツール

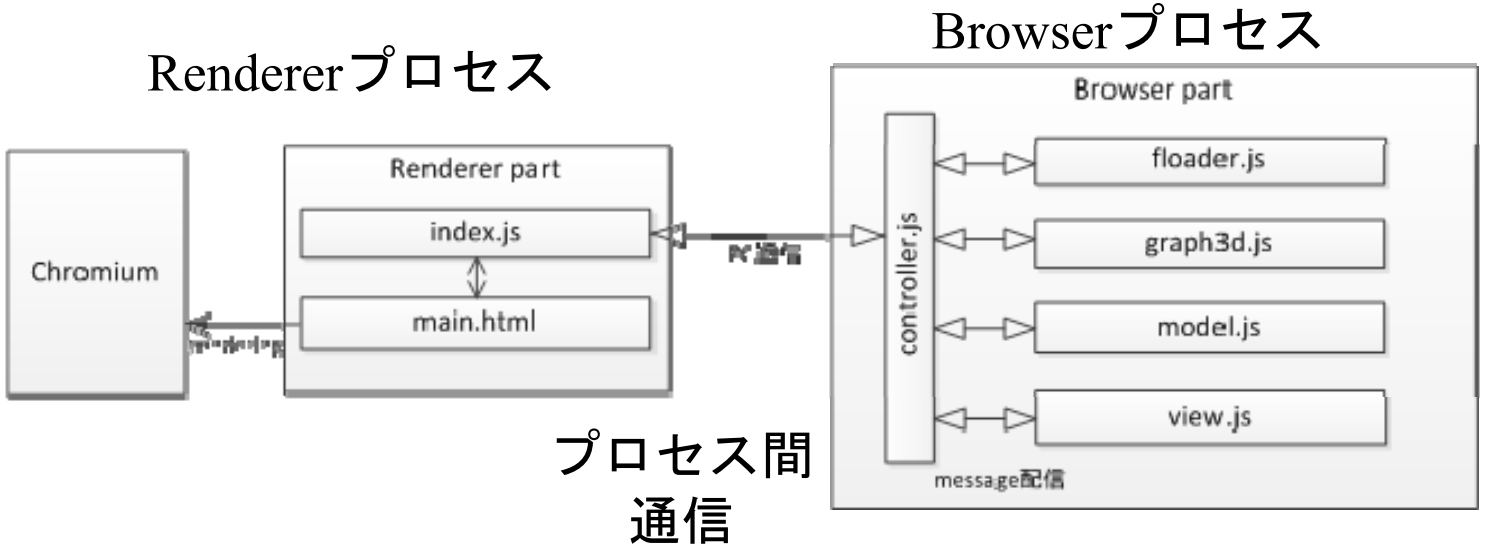
アプリケーションのデバッグ実行

1. Windows OSの場合，コマンドプロンプト上から，ソースコード一式の「fistrprepost-project」フォルダの中にある，「fistrprepost」フォルダに移動

2. 「electron」コマンドを実行

> cd <path_to_project>¥fistrprepost-project¥fistrprepost
(ソースコードのフォルダへ移動)
> electron .
(引数に"."を指定)

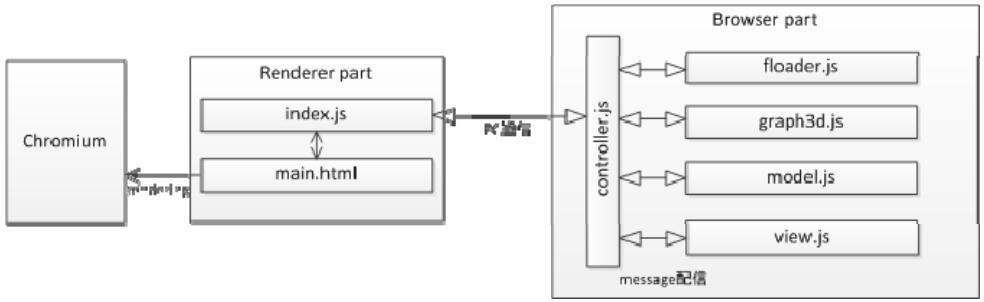
プログラムの構成 (1/2)



Renderer側：クライアントプログラム関連
ブラウザの中の描画処理：ブラウザ表示用のHTML (main.html) 及びそのActionを記述するJS (index.js)

Browser側：サーバープログラム関連
サーバーが行うGUIのパーツ処理のJS
(controller.js, floader.js, graph3d.js, model.js, view.js)

プログラムの構成 (2/2)



ディレクトリ				ファイル	説明
第1階層	第2階層	第3階層	第4階層		
				build.js	ビルド用JS
fistrprepost	js	etc		ConvexGeometry.js	
				jquery.exresize.0.1.0.js	
				jquery-1.7.1.min.js	jQuery
				OrbitControls.js	カメラ制御ライブラリ
				Projector.js	
				three.min.js	JavaScript 3D library
		wel	vm	floader.js	ファイル読み込み処理
				graph3d.js	３Dグラフ描画処理
				model.js	モデル処理
				view.js	ブラウザ表示処理
				s.cnt.template	解析制御データ雛形ファイル（sa）
				t.cnt.template	解析制御データ雛形ファイル（ta）
				s.hecmw_ctrl.dat.template	全体制御データ雛形ファイル（sa）
				t.hecmw_ctrl.dat.template	全体制御データ雛形ファイル（ta）
				controller.js	コントローラ
	node_modules			electron-tree-view	electron-tree-view一式
	css			style.css	デザイン（スタイルシート）
	ext	win		libgcc_s_seh-1.dll	要素圧縮プログラム依存ライブラリ
				libstdc++-6.dll	
				libwinpthread-1.dll	
				msh2data.exe	要素圧縮プログラム本体
				index.js	ブラウザ連携処理
				main.html	ブラウザ表示HTML
				welding.ico	プログラムアイコン

要素タイプの追加

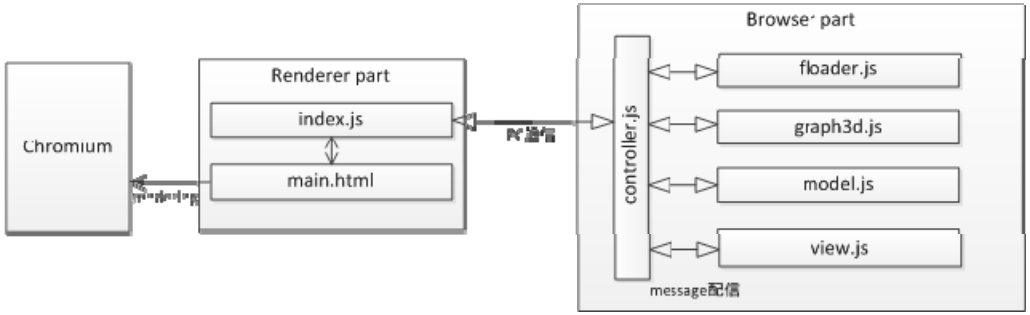
model.js

- 本システムのモデルの動作が集約
- Modelという基底クラス (モデルを抽象化した動作を定義している) が用意

本システムに新たなモデル (要素タイプ) を追加する手順

1. Model基底クラスを派生させた具象クラスを定義する (「Model抽象化クラス抽象関数」を新たなモデル用にオーバライドする)
2. プログラムイベント「create-model」要求時にそのインスタンスを返す

イベント（解析制御ファイル設定）の追加



part	送受	方向	実装例
Renderer part	送信	Ri⇒Bc	mw.webContents.send('action', {message: "close", arg: null});

Browser part	受信	Ri⇒Bc	ipcRenderer.on(
			'action',
			function(event, action) {
			dispachEevent(action.message, action.arg);
			}
			);

	送信	Bc⇒Bx	event.emit(message, args);

	受信	Bx⇒Bx	event.on(
			'init-resource',
			function(arg) {
			resource = arg.res;
			}
			);

	送信	Bx⇒Bx	dispachEvent('set-cursor', "wait");

Ri：ブラウザ連携処理 (index.js)

Bc：コントローラ (controller.js)

Bx：機能モジュール (floader.js, graph3d.js, model.js, view.js)

mw：BrowserWindow

event：機能モジュールのEventEmitter

多言語への対応

No.	名称	ファイル	ディレクトリ	locale	説明
1	ブラウザ表示HTML	main_(*).html	fistrprepost	jp	U/Iデザイン格納
2	言語リソース	resource_(*).js	fistrprepost	jp	言語リソース格納

(*)：ロケール（言語名）

別言語への変更

- 別の言語依存リソースを用意 (ファイル名の一部を別のロケールで置き換えたもの) を準備する
- 別のロケールをプログラム実行時の引数として指定する

> simwelding.exe --lang=(*)

(--lang=(*)なしならば, (*)はjpとなる)