Examples Manual

Release 1.0.0

Morphorm

Jan 18, 2025

INTRODUCTION

1 Introduction **3**
- 1.1 Description . . . 3
- 1.2 Workflow . . . 4
- 1.3 User Support . . . 4

2 Elliptic Physics **5**
- 2.1 Thermal Physics . . . 5
  - 2.1.1 Example 1: Convection . . . 5
- 2.2 Mechanical Physics . . . 8
  - 2.2.1 Example 1: Beam with Hole . . . 8
- 2.3 Thermomechanical Physics . . . 10
  - 2.3.1 Example 1: Bracket . . . 11
- 2.4 Electrical Physics . . . 14
  - 2.4.1 Example 1: Copper Wire . . . 14
- 2.5 Electrothermal Physics . . . 18
  - 2.5.1 Example 1: Joule Heating . . . 19
- 2.6 Nonlinear Mechanical Physics . . . 23
  - 2.6.1 Example 1: Cantilever Beam . . . 23
- 2.7 Nonlinear Thermomechanical Physics . . . 26
  - 2.7.1 Example 1: Cantilever Beam . . . 26

3 Topology Optimization **31**
- 3.1 Description . . . 31
- 3.2 Mechanical . . . 31
  - 3.2.1 Example 1: Compliance Minimization . . . 31
  - 3.2.2 Example 2: Prune & Refine . . . 36
  - 3.2.3 Example 3: Local Stress Constraints . . . 41
  - 3.2.4 Example 4: Center of Gravity . . . 45
  - 3.2.5 Example 5: Inverse Criterion . . . 51
  - 3.2.6 Example 6: Multi-Level Optimization . . . 58
- 3.3 Thermal . . . 67
  - 3.3.1 Example 1: Compliance Minimization . . . 67
  - 3.3.2 Example 2: Local Thermal Flux Constraints . . . 72
- 3.4 Thermomechanics . . . 76
  - 3.4.1 Example 1: Compliance Minimization . . . 76
  - 3.4.2 Example 2: Multi-Dimensional Parameter Study . . . 84
  - 3.4.3 Example 3: Local Stress Constraints . . . 91
  - 3.4.4 Example 4: Multi-Scenario Optimization . . . 97
- 3.5 Nonlinear Mechanics . . . 103
  - 3.5.1 Example 1: Maximize Deformations . . . 103

4 **Shape Optimization** **111**
4.1 Description . . . 111
4.2 Example 1: Gradient-Based Optimization . . . 111
4.2.1 Model Definition . . . 112
4.2.2 Formulation . . . 112
4.2.3 Input Deck . . . 112
4.2.4 Run Study . . . 115
4.2.5 Results . . . 115
4.3 Example 2: Surrogate-Based Optimization . . . 116
4.3.1 Input Deck . . . 116
4.3.2 Run Study . . . 122
4.3.3 Results . . . 122

Bibliography **125**

These pages are meant to serve as the `Examples Manual` for the Morphorm software.

Note

CHAPTER

ONE

INTRODUCTION

The *Introduction* chapter briefly describes the Morphorm engineering design modules.

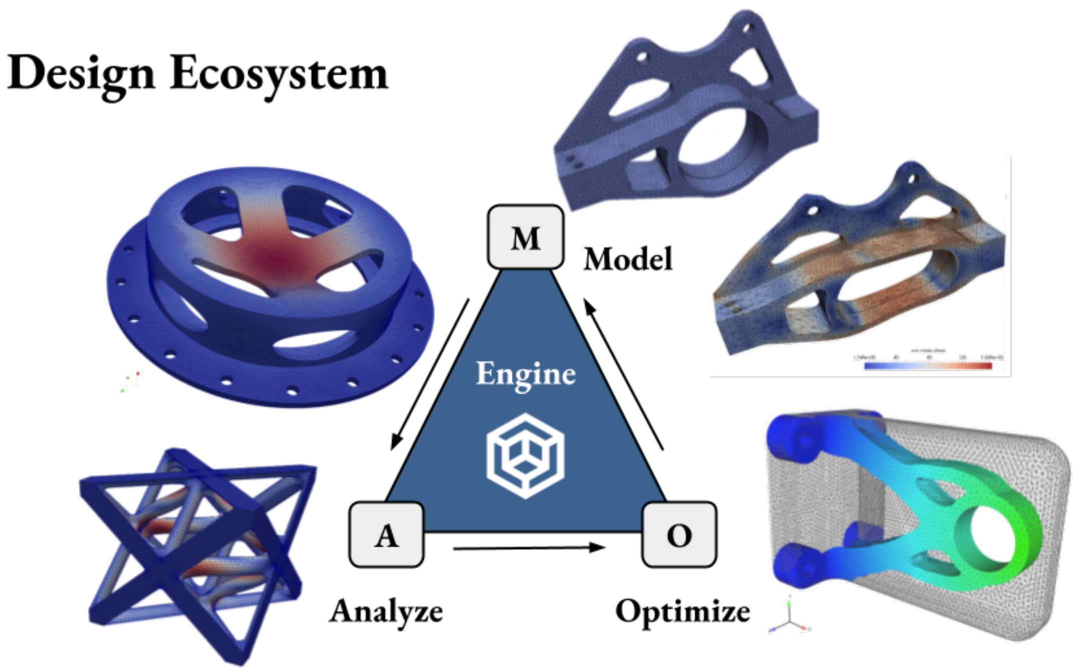

Fig. 1.1: The Morphorm® design ecosystem is comprised of four modules: (1) Analyze, (2) Model, (3) Optimize, and (4) Engine. The Analyze module predicts how a physical system interacts with its environment by numerically solving the partial differential equations governing the behavior of the system. The Model module constructs the geometry model describing the physical system under investigation. The Optimize module applies optimization methods to improve the performance, reliability, and efficiency of the physical system. The Engine module manages the exchange of information between the analysis, modeling, and optimization modules, including other third-party applications. If multiple environments are considered in a design study, the Engine applies a Multiple-Program, Multiple-Data parallel programming model to enable concurrent evaluations of the physical systems.

1.1 Description

Morphorm® builds modeling, simulation, and optimization technologies to deliver real-time engineering. Our mission is to employ advanced scientific computing to solve complex science and engineering problems. We aspire to be the most innovative computational engineering company, where our partners can discover in real-time optimized design solutions.

Our products exploit distributed memory parallel programming models and graphical processing units (GPUs) to accelerate engineering discovery. The Morphorm platform shown in Fig. 1.1 includes tools for multiphysics finite element analysis, design optimization, geometry modeling, uncertainty quantification and propagation, machine learning, and much more. Explore our software and accelerate product discovery.

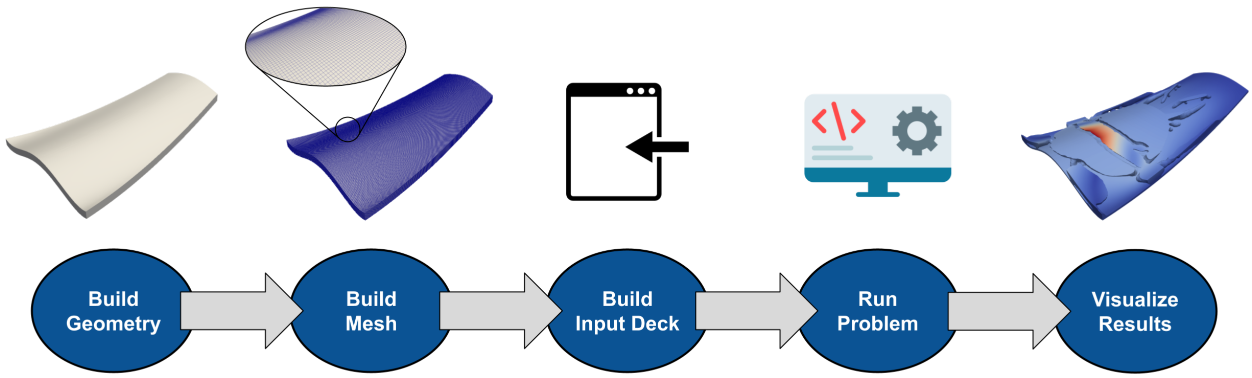

Fig. 1.1.1: Pictorial description of the Morphorm design workflow.

1.2 Workflow

The sequence of steps to setup and run a Morphorm study are described in Fig. 1.1.1. Setting up and running a Morphorm study is done through a sequence of steps, which is refer in this manuscript as a design workflow. The design workflow consists of the following steps:

- Construct the geometry
- Create the exodus mesh [Sjaardema]
- Write the input deck
- Launch the design study
- Postprocess the results

The initial geometry and exodus mesh files needed to run a Morphorm study are currently created outside the Morphorm environment. For example, the Cubit® geometry and mesh toolkit can be used to create the initial geometry and exodus mesh files. If the user has an existing geometry file, the geometry file can be uploaded to Cubit® to create the exodus mesh file. Cubit® also supports other mesh file formats for many commercial-off-the-shelf engineering software tools. For instance, an Abaqus input file can be converted into an exodus mesh file in Cubit®. The user is recommended to consult the Cubit user manual for more information about the mesh and geometry file formats supported in Cubit [Stimpson *et al.*]. The results from a Morphorm design study can be visualized and postprocessed with ParaView or any other commercial-off-the-shelf visualization tool that supports the exodus mesh file format. The `Morphorm User Manual` gives an in-depth presentation of the Morphorm input deck.

1.3 User Support

You can submit questions via email to help@morphorm.com.

CHAPTER

TWO

ELLIPTIC PHYSICS

The Elliptic Physics chapter provides examples describing how to setup and simulate elliptic physics.

2.1 Thermal Physics

The following examples demonstrate how to use the Analyze module to simulate thermal elliptic physics. The examples presented herein create a suite of tutorials which summarize the main concepts for simulating thermal elliptic physics.

2.1.1 Example 1: Convection

This example shows a two-dimensional steady-state thermal analysis including convection to a prescribed external (ambient) temperature. The example is taken from the NAFEMS benchmark collection.

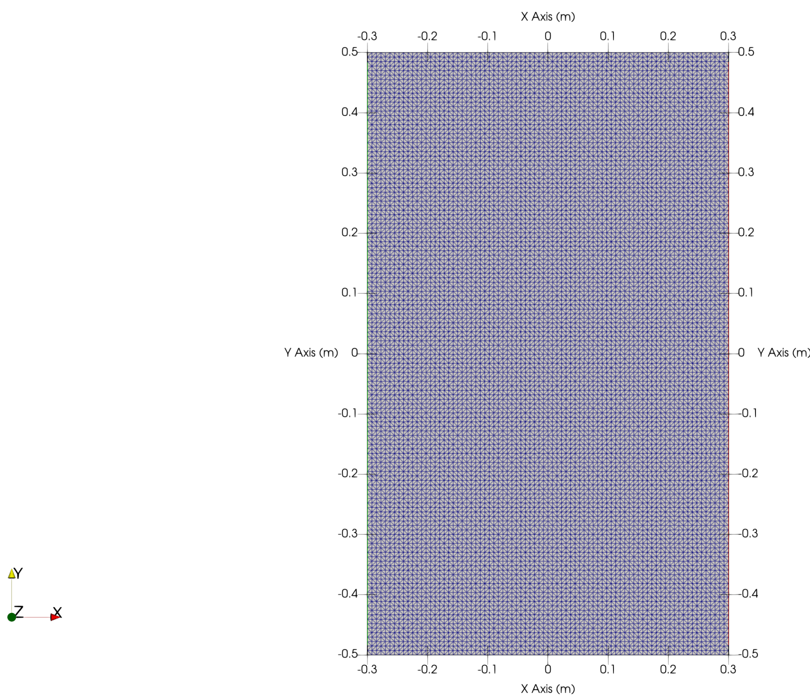

Fig. 2.1.1: The computational domain and finite element mesh used for the thermal simulation. The lower wall is kept at $100°$C. The left wall is insulated. The upper and right walls are convecting to $0°$C.

Model Definition

Fig. 2.1.1 shows the computational domain and the finite element mesh used for the thermal simulation. The thermal simulation aims to find the temperature distribution in the computational domain shown in Fig. 2.1.1. The boundary conditions for the simulation are:

1. Dirichlet boundary conditions
   a. The lower wall (plane at $Y = -0.5$) is kept at $100°$C.
2. Neumann boundary conditions
   a. The left wall (plane at $X = -0.3$) is insulated, i.e., the heat flux is zero.
   b. The upper (plane at $Y = 0.5$) and right (plane at $X = 0.3$) walls are convecting to $0°$C with a heat transfer coefficient of $750\ \mathrm{W}/(\mathrm{m}^2 \cdot^\circ \mathrm{C})$.

The material conductivity constant is set to $52\ \mathrm{W}/(\mathrm{m} \cdot^\circ \mathrm{C})$.

Input Deck

The input deck setup for the thermal analysis is:

Listing 2.1.1: Input deck setup for the simulation of the thermal elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
  update_problem true
  update_problem_frequency 1
end service

begin scenario 1
  physics steady_state_thermal
  service 1
  dimensions 2
  loads 1 2
  boundary_conditions 1
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  output_data true
  data temperature temperature_gradient thermal_flux
  native_service_output true
  output_frequency 1
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name temp
  degree_of_freedom temp
```

(continues on next page)

(continued from previous page)

```
  value 100.
end boundary_condition

begin load 1
  type thermal_flux
  location_type sideset
  location_name insulation
  value 0.
end load

begin load 2
  type thermal_convection
  location_type sideset
  location_name convective_flux
  value 750.
  material_id 1
end load

begin block 1
  material 1
  name body
end block

begin material 1
  material_model isotropic_linear_thermal
  thermal_conductivity 52.
  reference_temperature 0.
end material

begin study
  method analysis
end study

begin mesh
  name elliptic_thermal_ex1.exo
end mesh
```

Run Study

To run the thermal simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_thermal_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `elliptic_thermal_ex1.i` is the *input deck* for the thermal simulation. The simulation should run after pressing the `Enter` key.

Results

The benchmark result for the target location at $(0.3\text{m}, -0.3\text{m})$ is a temperature of $18.25°\text{C}$. The thermal simulation with the mesh of linear triangles shown in Fig. 2.1.1 gives a temperature of $18.249°\text{C}$ at $(0.3\text{m}, -0.3\text{m})$. The temperature field for the steady state thermal analysis is shown in Fig. 2.1.2. The expected temperature distribution is computed by the thermal solver.

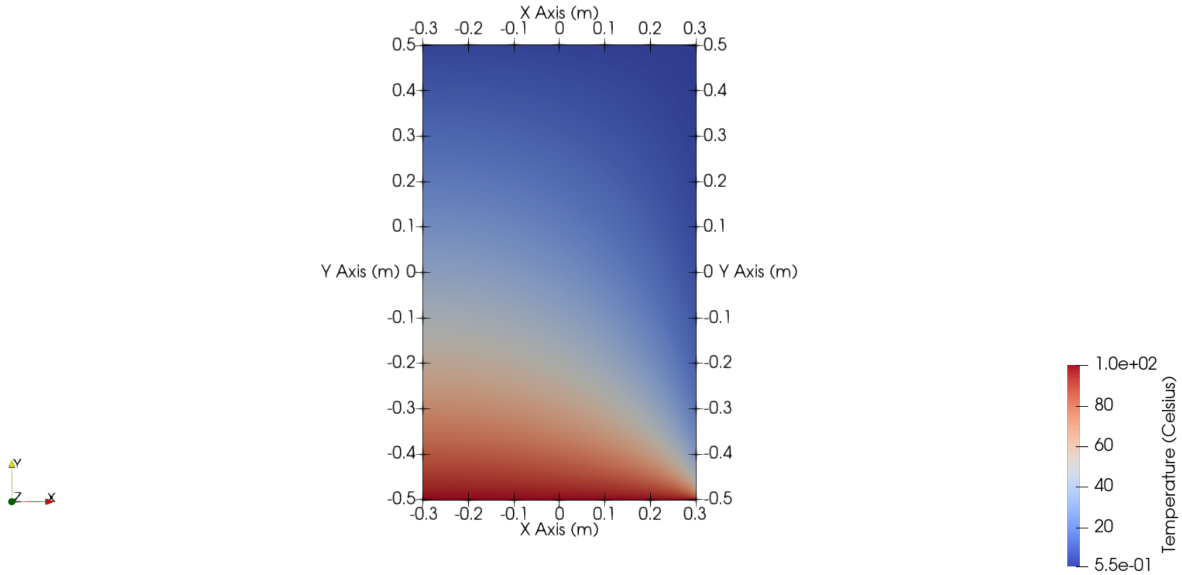

Fig. 2.1.2: Temperature field for the steady state thermal analysis. The temperature at $(0.3\text{m}, -0.3\text{m})$ was measured at $18.249°\text{C}$. The expected temperature at $(0.3\text{m}, -0.3\text{m})$ is $18.25°\text{C}$.

2.2 Mechanical Physics

The following examples demonstrate how to use the Analyze module to simulate mechanical elliptic physics. The examples create a suite of tutorials which summarize the main concepts for simulating mechanical elliptic physics.

2.2.1 Example 1: Beam with Hole

This example describes how to set up and run a linear mechanics analysis of a beam with a hole at the center. The beam is fixed at one end and loaded at the opposite end.

Model Definition

Fig. 2.2.1 shows the computational domain and the finite element mesh used for the linear mechanical analysis. The mechanical simulation aims to find the displacement and stress distributions in the computational domain shown in Fig. 2.2.1. The boundary conditions for the linear mehcanical simulation are:

1. Dirichlet boundary conditions

   a. The X, Y, and Z displacements (in) on the surface defined by the plane at $X = 0.0$ are fixed.

2. Neumann boundary conditions

   a. A traction force (lbf) with components $(224809.0, 0.0, 0.0)$ is applied on the green surface defined by the plane at $X = 2.0$.

The modulus of elasticity is set to 29000 ksi and Poisson's ratio is set to 0.25.

Input Deck

The input deck setup for the linear mechanical analysis is:

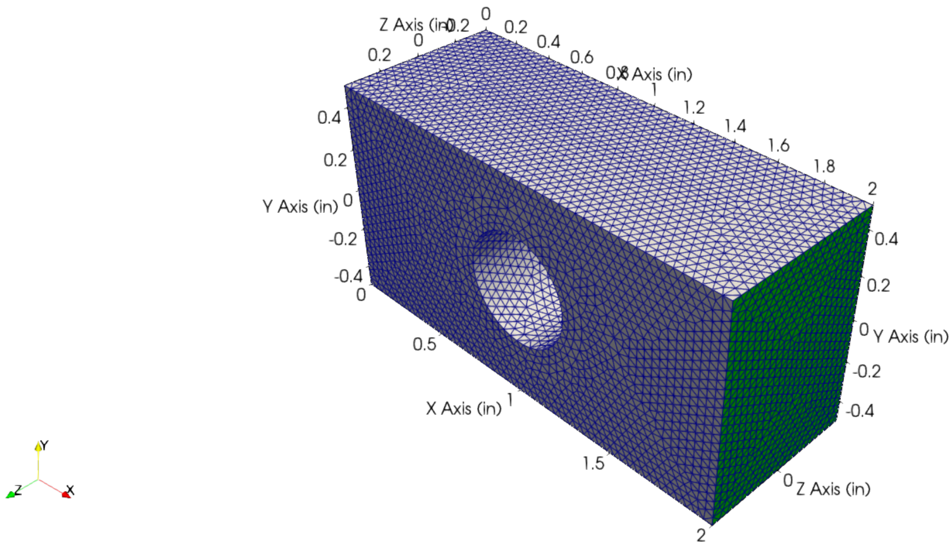

Fig. 2.2.1: The computational domain and the finite element mesh used for the linear mechanical simulation. The X, Y, and Z displacements (in) on the surface defined by the plane at $X = 0.0$ are fixed. A traction force (lbf) with components $(224809.0, 0.0, 0.0)$ is applied on the green surface define by the plane at $X = 2.0$.

Listing 2.2.1: Input deck setup for the simulation of the linear mechanical elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
end service

begin scenario 1
  physics steady_state_mechanics
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  data dispx dispy dispz vonmises
  native_service_output true
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
```

(continues on next page)

(continued from previous page)

```
  location_name ns_1
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name ss_1
  value 224809 0 0
end load

begin block 1
  material 1
  name block_1
end block

begin material 1
  material_model isotropic_linear_elastic
  poissons_ratio 0.25
  youngs_modulus 29000000
end material

begin study
  method analysis
end study

begin mesh
  name elliptic_mechanical_ex1.exo
end mesh
```

Run Study

To run the simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_mechanical_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `elliptic_mechanical_ex1.i` is the *input deck* for the linear mechanical analysis. The simulation should run after pressing the `Enter` key.

Results

The displacement (in) and von Mises stress (psi) fields are shown in Fig. 2.2.2. The expected displacement and von Mises stress fields are computed. As expected, the largest displacement occur on the surface defined by the plane at $X = 2$ while the largest von Mises stress is observed at the center (plane at $X = 1$) of the beam near the hole.

2.3 Thermomechanical Physics

The following examples demonstrate how to use the Analyze module to simulate thermomechanical elliptic physics. The examples create a suite of tutorials which summarize the main concepts for simulating thermomechanical elliptic physics.

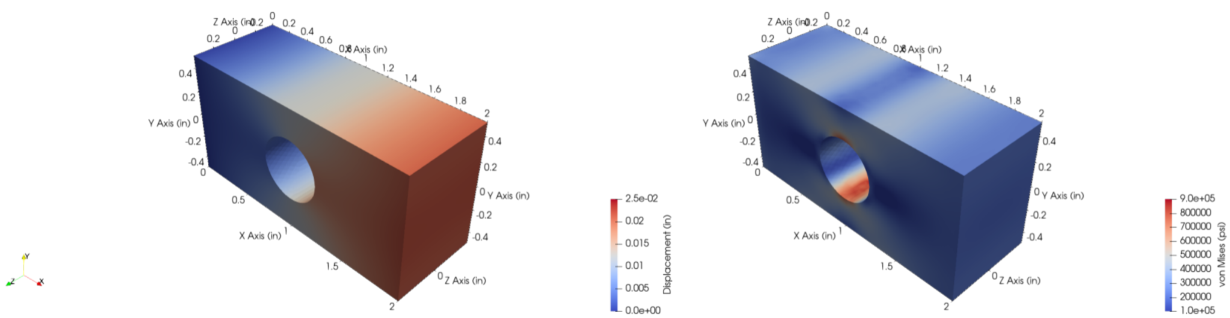

Fig. 2.2.2: The displacement (left pane) and von Mises stress (right pane) fields. The largest displacement is observed on the surface defined by the plane at $X = 2$ while the largest von Mises stress is observed at the center (plane at $X = 1$) of the beam near the hole.

2.3.1 Example 1: Bracket

This example considers the thermomechanical analysis of a critical aicraft bracket. The bracket is modeled as a three-dimensional geometry. Loading brackets on a jet engine play a very critical role. These structures must support the weight of the engine during handling without breaking or warping. The brackets may be used only periodically, but they stay on the engine at all times, including during flight. The goal in this example is to investigate the displacement, temperature, and von Mises stress distributions produced by external forces.

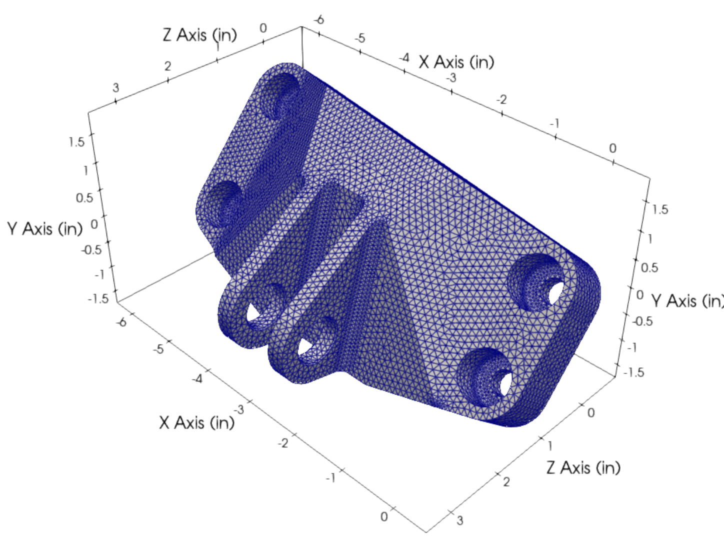

Fig. 2.3.1: The computational domain and the finite element mesh used for the thermomechanical analysis.

Model Definition

Fig. 2.3.1 shows the computational domain and the finite element mesh used for the thermomechanical analysis. The simulation aims to find the temperature (°F), displacement (in), and von Mises stress (psi) distributions due to thermal and mechanical loading. The boundary conditions for the thermomechanical analysis are:

1. Dirichlet boundary conditions

   a. The X, Y, and Z displacements (in) on the pink surfaces are fixed.

   b. The cyan surface is kept at $75.0°$F.
   c. The purple surfaces is kept at $100.0°$F.
2. Neumann boundary conditions
   a. A traction force with components $(0.0, 0.0, 8000)$ lbf is applied to the purple surfaces shown in Fig. 2.3.2

The modulus of elasticity is set to 29000 ksi and Poisson's ratio is set to 0.29. The material conductivity constant is set to 112.0 BTU-in$/$(hr-ft^2 $-°$ F) or 2.017 lbf$/$(s-$°$F). The coefficient of thermal expansion is set to 9.61 μin (in-$°$F) and the reference temperature is set to $0.0°$F.

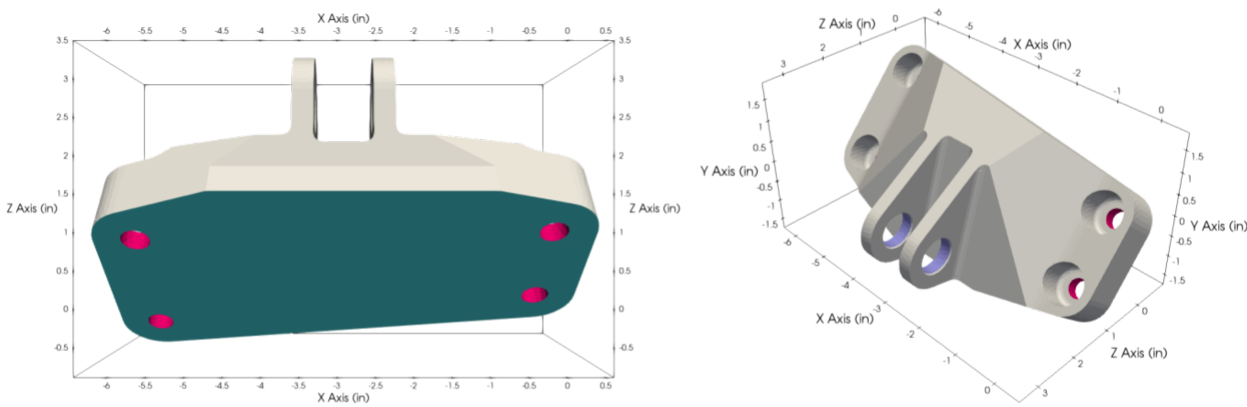

Fig. 2.3.2: This figure shows the thermal and mechanical boundary conditions used for the thermomechanical analysis. The X, Y, and Z displacements (in) on the pink surfaces are fixed. The cyan surface is kept at $75.0°$F. The purple surfaces are kept at $100.0°$F. A traction force (lbf) with components $(0.0, 0.0, 8000.0)$ is applied on the purple surfaces.

Input Deck

The input deck setup for the thermomechanical analysis is:

Listing 2.3.1: Input deck setup for the simulation of the linear thermomechanical elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
end service

begin scenario 1
  physics steady_state_thermomechanics
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1 2 3
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  data dispx dispy dispz temperature vonmises
```

(continues on next page)

(continued from previous page)

```
  native_service_output true
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name bolts
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name base
  degree_of_freedom temp
  value 75.
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
  location_name clamps
  degree_of_freedom temp
  value 100.
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name load
  value 0 0 8e3
end load

begin block 1
  material 1
  name block_1
end block

begin material 1
  material_model isotropic_linear_thermoelastic
  poissons_ratio 0.342
  youngs_modulus 16500e3
  thermal_expansivity 4.78e-6
  thermal_conductivity 0.8376
  reference_temperature 0.0
end material

begin study
  method analysis
end study
```

(continues on next page)

(continued from previous page)

```
begin mesh
  name elliptic_thermomechanical_ex1.exo
end mesh
```

Run Study

To run the thermomechanical simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_thermomechanical_ex1.i
```

`runmorphorm.py` is the python script with the sequence of instructions to run the Morphorm software and `elliptic_thermomechanical_ex1.i` is the *input deck* for the thermomechanical simulation. The simulation should run after pressing the `Enter` key.

Results

The displacement, von Mises stress, and temperature fields for the aircraft bracket are shown in Fig. 2.3.3. Fig. 2.3.3 shows that the largest von Mises stresses are observed near the fixed bolt locations and where the traction forces are applied. Fig. 2.3.3 also shows that most of the loading bracket is not under substantial stress. Therefore, we can consider redesigning the bracket to reduce material usage while satisfying the performance requirements. This design problem will be investigated in the *Topology Optimization* chapter.

2.4 Electrical Physics

The following examples demonstrate how to use the Analyze module to simulate electrical elliptic physics. The examples create a suite of tutorials which summarize the main concepts for simulating electrical elliptic physics.

2.4.1 Example 1: Copper Wire

A current is created when a voltage difference is applied to a conductor. The relationship between the voltage ϕ, the electric current I, and the electrical resistance R in an electric circuit is described by Ohm's Law:

$$I = \phi/R. \tag{2.4.1}$$

Eq.2.4.1 shows that the current is directly proportional to the voltage. The proportionality constant is the resistance of the device. This example shows how to compute the voltage distribution, i.e., electric potential, of a copper wire. An isotropic material constitutive model is considered. Therefore, the second order electric conductivity tensor is defined as:

$$\bar{\sigma}_{ij} = \bar{\sigma} I_{ij}, \tag{2.4.2}$$

where I_{ij} is the identity tensor and, $\bar{\sigma} = 5.98e^7$ S/m is the electrical conductivity constant of copper at 20°C.

Model Definition

The 10 mm long copper wire shown in Fig. 2.4.1 with a 0.5 mm radius is considered. A constant electric current source of 1.0A is passed through the wire and the voltage drop is measured. The boundary conditions represent a connection to a direct current source of electric current. The surface defined by the plane at $Z = -5$ mm is grounded, representing an electric current sink, i.e., $I = 0$. The surface defined by the plane at $Z = 5$ mm is connected to a constant current source of 1 A. The outer surface is insulated, i.e., the electric flux is zero. The simulation aims to find the voltage distribution by solving Maxwell's equation of conservation of charge. The boundary conditions for the simulation are:

1. Dirichlet boundary conditions

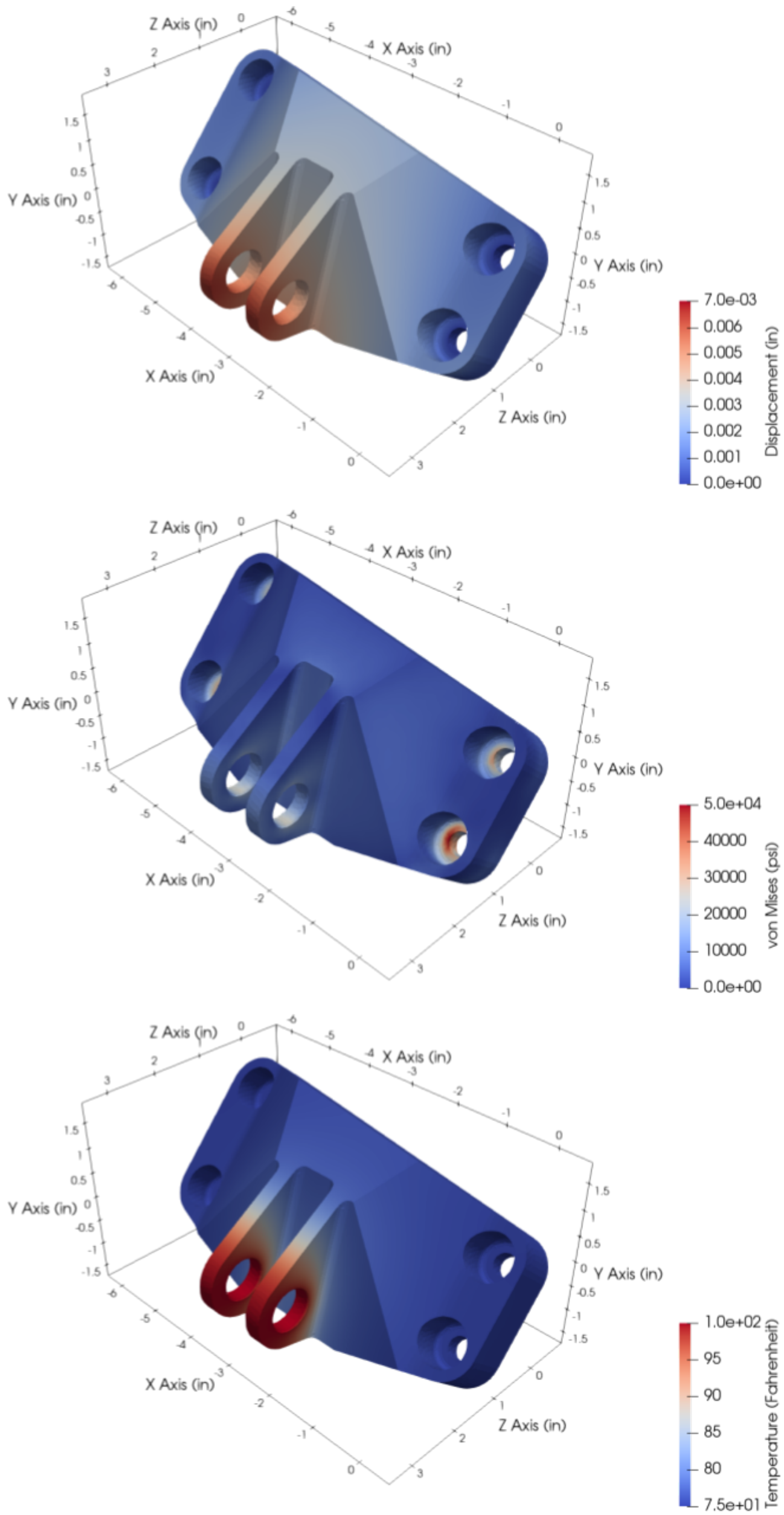

Fig. 2.3.3: The displacement (top pane), von Mises stress (middle pane), and temperature (bottom pane) fields for the aircraft bracket.

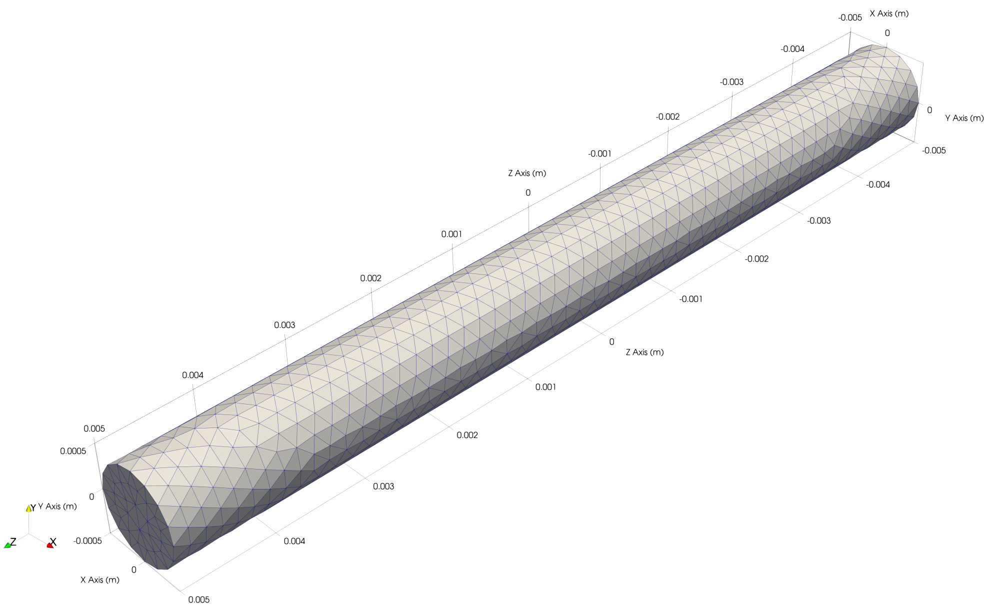

Fig. 2.4.1: The computational domain and the finite element mesh used for the copper wire simulation. The surface defined by the plane at $Z = -5$ mm is grounded. The surface defined by the plane at $Z = 5$ mm is connected to a constant current source of 1 A. The outer surface is insulated.

   a. The surface defined by the plane at $Z = -5$ mm is grounded.
   b. The surface defined by the plane at $Z = 5$ mm is connected to a constant current source of 1 A.

2. Neumann boundary conditions
   a. The outer surface is insulated.

The electrical resistance of a 10 mm long copper wire with a 0.5 mm radius is 0.212Ω. Therefore, the Dirichlet boundary conditions can be redefined as:

- $\phi = 0$ V on the surface defined by the plane at $Z = -5$ mm, and
- $\phi = 0.212$ V on the surface defined by the plane at $Z = 5$ mm,

where Eq.2.4.1 was used to solve for the electric potential.

Input Deck

The input deck setup for the electrical simulation is:

Listing 2.4.1: Input deck setup for the simulation of the electrical elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
  update_problem true
  update_problem_frequency 1
end service
```

(continues on next page)

(continued from previous page)

```
begin scenario 1
  physics steady_state_electrical
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1 2
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  data electric_potential
  native_service_output true
  output_frequency 1
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name bottom
  degree_of_freedom potential
  value 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name top
  degree_of_freedom potential
  value 0.212
end boundary_condition

begin load 1
  type electrical_flux
  location_type sideset
  location_name insulation
  value 0.
end load

begin block 1
  material 1
  name body
end block

begin material 1
  material_model isotropic_electrical
  electrical_conductivity 5.998e7
  electrical_resistance 0.212e-3
end material
```

(continues on next page)

(continued from previous page)

```
begin study
  method analysis
end study

begin mesh
  name elliptic_electrical_ex1.exo
end mesh
```

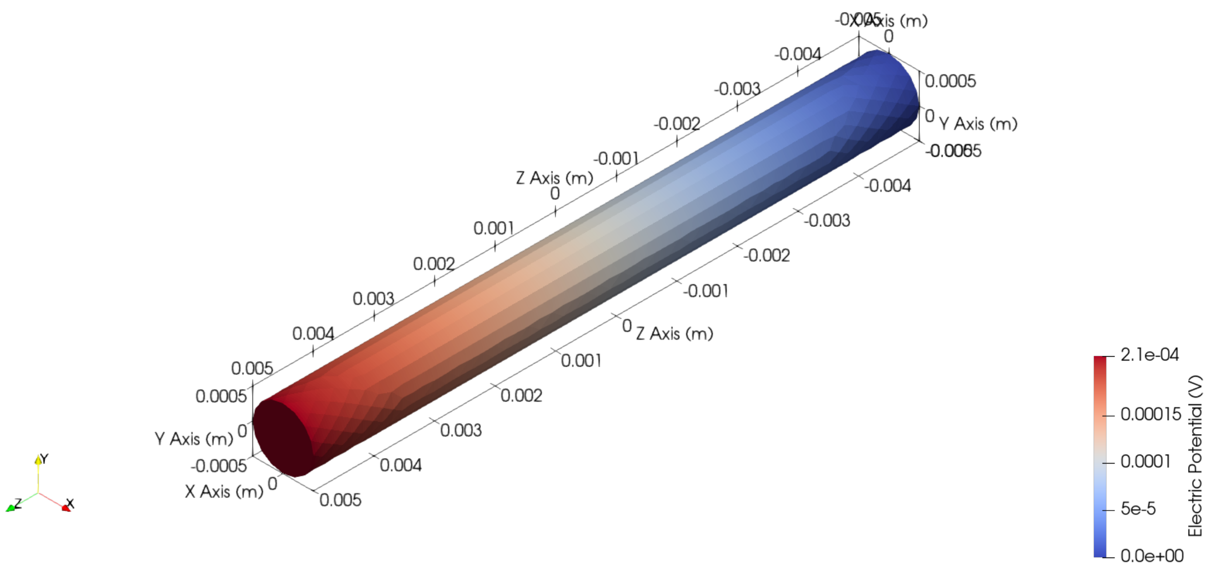

Fig. 2.4.2: Electric potential (voltage) field along the copper wire.

Run Study

To run the electrical simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_electrical_ex1.i
```

`runmorphorm.py` is the python script containing the sequence of instructions to run the Morphorm software and `elliptic_electrical_ex1.i` is the *input deck* for the electrical simulation. The simulation should run after pressing the `Enter` key.

Results

The voltage distribution is plotted in Fig. 2.4.2. The expected linear voltage drop is observed along the copper wire.

2.5 Electrothermal Physics

The following examples demonstrate how to use the Analyze module to simulate electrothermal elliptic physics. These examples create a suite of tutorials which summarize the main concepts for simulating electrothermal elliptic physics.

2.5.1 Example 1: Joule Heating

A Joule heat source, defined by Joule's first law, gives the relationship between the heat generated and the current flowing through a conductor. Mathematically, a Joule heat source is defined as:

$$Q^J = J_i \phi_i \quad \text{for} \quad i = 1, \ldots, d \tag{2.5.1}$$

where ϕ_i is the electric potential and d is the spatial dimension. The current density is defined as:

$$J_i = \bar{\sigma}_{ij} \phi_j = \bar{\sigma} I_{ij} \quad \text{for} \quad i, j = 1, \ldots, d \tag{2.5.2}$$

The material electrical conductivity constant is defined by $\bar{\sigma} > 0$ and I_{ij} is the second-order identity tensor. The Joule heat source term is coupled to the steady state heat conduction equation as a thermal source term.

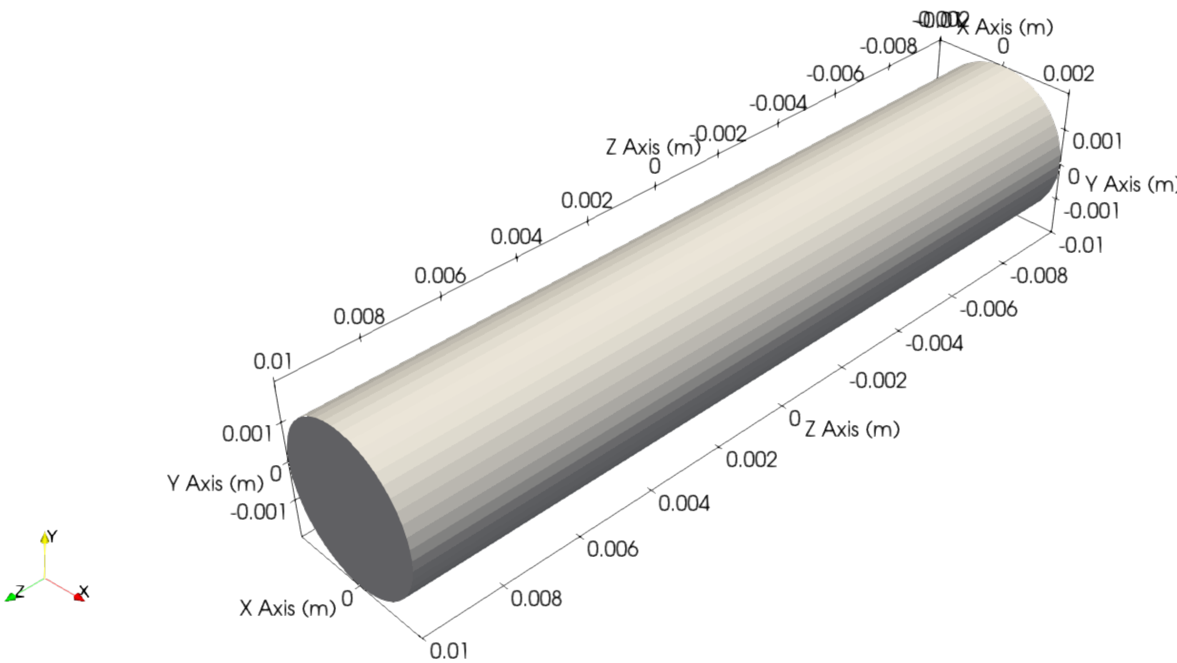

Fig. 2.5.1: The computational domain and the finite element mesh for the copper wire simulation. The surface defined by the plane at $Z = -0.01$ (bottom surface) is grounded, i.e., the electric potential is zero. An electric potential of 50mV is applied to the surface defined by the plane at $Z = 0.01$ (top surface). The temperature at the bottom and top surfaces is kept at 295 K. A Joule heating source is modeled along the copper wire.

Model Definition

Joule heating of a resistor is studied in this example. The 2.0 cm long copper wire shown in Fig. 2.5.1 with a 0.2 cm radius is investigated. The objective is to find the electric potential and temperature distributions in the copper wire.

The boundary conditions for the simulation are:

1. Dirichlet boundary conditions

   a. The surface defined by the plane at $Z = -0.01$ (bottom surface) is grounded, i.e., the electric potential is zero.

   b. An electric potential of 50mV is applied to the surface defined by the plane at $Z = 0.01$ (top surface).

   c. The temperature at the bottom and top surfaces is kept at 295 K.

2. Volume force

   a. A Joule heat source is modeled.

The material electrical conductivity constant is $5.998e^7$ S/m. The material thermal conductivity constant is 400 $\mathrm{Wm^{-1}K^{-1}}$ and the reference temperature is set to 298 K.

Input Deck

The input deck setup for the electrothermal simulation is:

Listing 2.5.1: Input deck setup for the simulation of the electrothermal elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
  update_problem true
  update_problem_frequency 1
end service

begin scenario 1
  physics steady_state_electrothermal
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1 2 3 4
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  native_service_output true
  data temperature temperature_gradient thermal_flux
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name top
  degree_of_freedom temp
  value 295
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type sideset
  location_name bottom
  degree_of_freedom temp
  value 295
end boundary_condition

begin boundary_condition 3
```

(continues on next page)

(continued from previous page)

```
  type fixed_value
  location_type sideset
  location_name top
  degree_of_freedom potential
  value 50e-3
end boundary_condition

begin boundary_condition 4
  type fixed_value
  location_type sideset
  location_name bottom
  degree_of_freedom potential
  value 0
end boundary_condition

begin load 1
  type joule_heating
  location_name body
end load

begin block 1
  material 1
  name body
end block

begin material 1
  material_model isotropic_electrothermal
  thermal_conductivity 400
  reference_temperature 298
  electrical_resistance 0.212e-3
  electrical_conductivity 5.998e7
end material

begin study
  method analysis
end study

begin mesh
  name elliptic_electrothermal_ex1.exo
end mesh
```

Run Study

To run the electrothermal simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_electrothermal_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `elliptic_electrothermal_ex1.i` is the *input deck* for the electrothermal simulation. The analysis should run after pressing the `Enter` key.

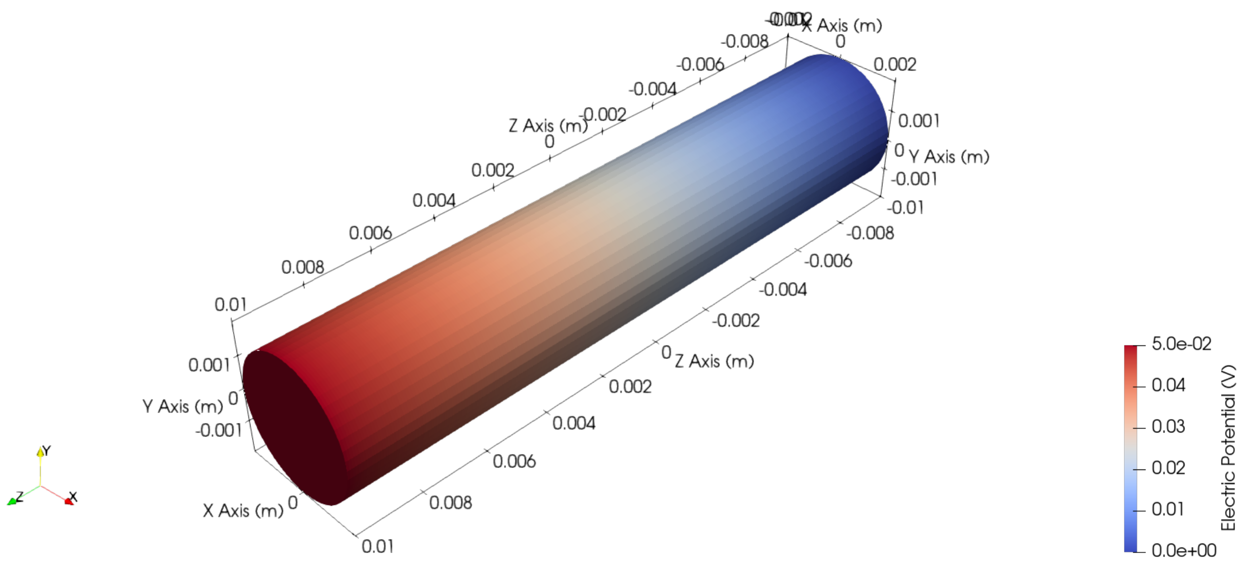

Fig. 2.5.2: Electric potential distribution along the copper wire.

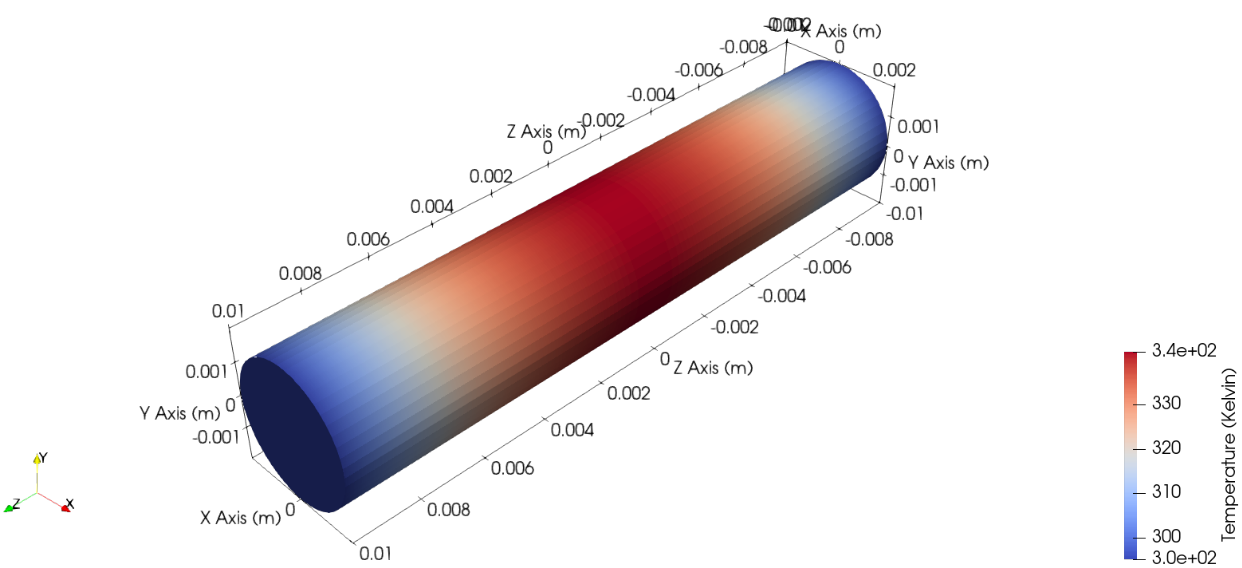

Fig. 2.5.3: Temperature distribution along the copper wire.

Results

The voltage and temperature fields for the copper wire are shown in Fig. 2.5.2 and in Fig. 2.5.3, respectively. The expected linear voltage drop is observed along the copper wire. The minimum temperature is observed at the surfaces defined by the planes at $Z = 0.01$ (top surface) and $Z = -0.01$ (bottom surface). The expected peak temperature of approximately 340 K is observed at the center of the copper wire [Joe *et al.*]. The 45.0 K increase in temperature is due to the 50.0 mV potential applied to the top surface.

2.6 Nonlinear Mechanical Physics

The following examples demonstrate how to use the Analyze module to simulate nonlinear mechanical elliptic physics. The examples create a suite of tutorials which summarize the main concepts for simulating nonlinear mechanical elliptic physics.

2.6.1 Example 1: Cantilever Beam

This example shows the three-dimensional nonlinear mechanical simulation of a cantilever beam. The cantilever beam is fixed at the surface defined by the plane at $X = -5.0$ and loaded at the surface defined by the plane at $X = 5.0$.

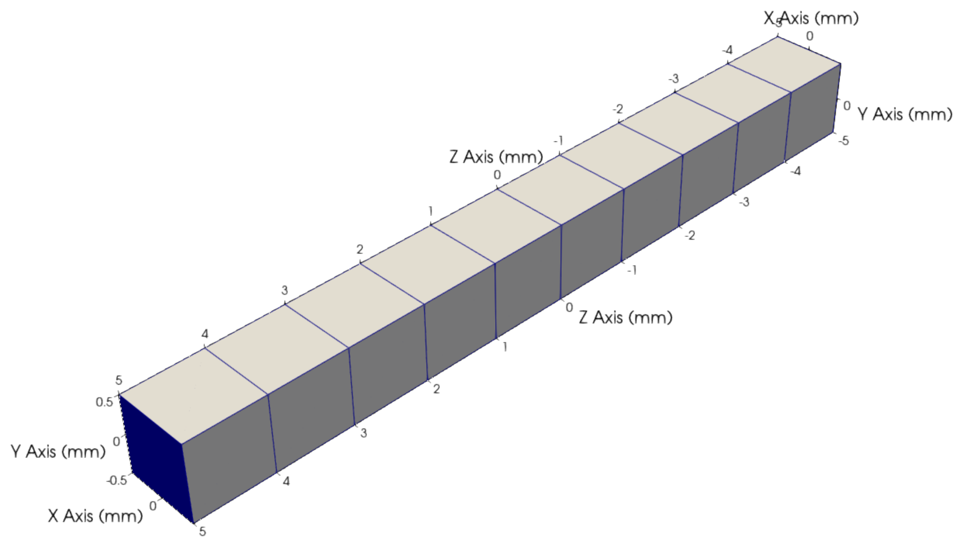

Fig. 2.6.1: The computational domain and the finite element mesh used for the nonlinear mechanical analysis. The X, Y, and Z displacements on the surface defined by the plane at $X = -5.0$ are fixed. A traction force with components $(0.0, 0.0, 5e6)$ is applied on the surface defined by the plane at $X = 5.0$.

Model Definition

Fig. 2.6.1 shows the computational domain and the finite element mesh used for the nonlinear mechanical analysis. The simulation aims to find the displacement distribution due to the external traction force. The Dirichlet and Neumann boundary conditions for the nonlinear mechanical simulation are:

1. Dirichlet boundary conditions

    a. The X, Y, and Z displacements (in) on the surface defined by the plane at $X = -5.0$ are fixed.

2. Neumann boundary conditions

a. A traction force (lbf) with components $(0.0, 0.0, 5e6)$ is applied on the surface defined by the plane at $X = 5.0$.

The material modulus of elasticity is set to 29000 ksi and Poisson's ratio is set to 0.29.

Input Deck

The input deck setup for the nonlinear mechanical simulation is:

Listing 2.6.1: Input deck setup for the simulation of the nonlinear mechanical elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
end service

begin scenario 1
  physics steady_state_nonlinear_mechanics
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  data dispx dispy dispz vonmises
  native_service_output true
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name fixed
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name load
  value 0 0 5e6
end load

begin block 1
  material 1
  name block_1
end block
```

(continues on next page)

(continued from previous page)

```
begin material 1
  material_model isotropic_hyperelastic_neohookean
  poissons_ratio 0.29
  youngs_modulus 29000e3
end material

begin study
  method analysis
end study

begin mesh
  name elliptic_nonlinear_mechanical_ex1.exo
end mesh
```

Run Study

To run the simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_nonlinear_mechanical_ex1.i
```

`runmorphorm.py` is the python script with the sequence of instructions to run the Morphorm software and `elliptic_nonlinear_mechanical_ex1.i` is the *input deck* for the nonlinear mechanical simulation. The simulation should run after pressing the `Enter` key.

Results

The displacement distribution for the cantilever beam is shown in Fig. 2.6.2. The expected displacement distribution is computed.

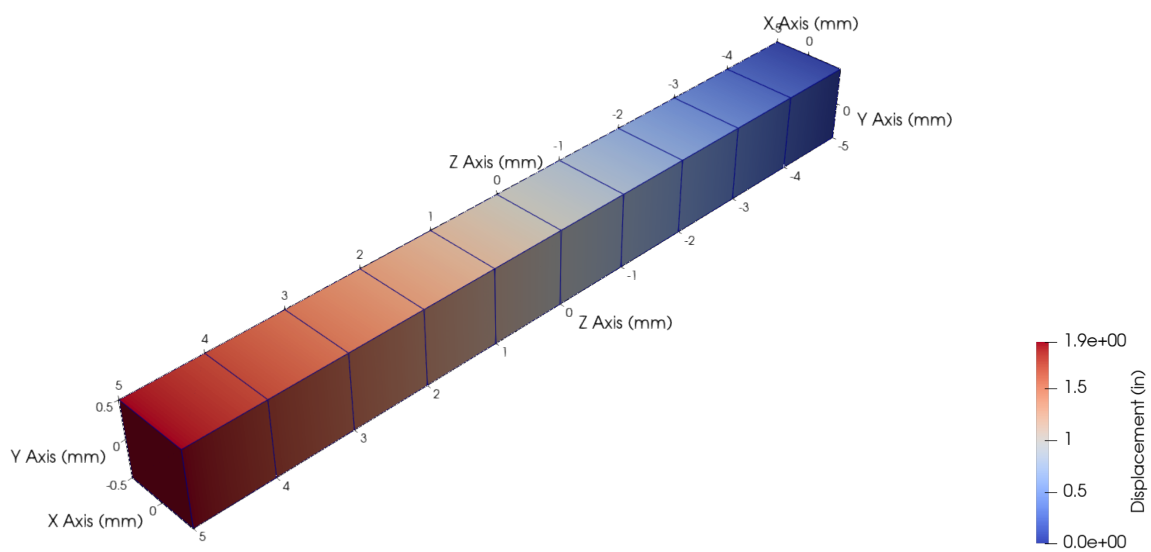

Fig. 2.6.2: The displacement distribution for the cantilever beam.

2.7 Nonlinear Thermomechanical Physics

The following examples demonstrate how to use the Analyze module to simulate nonlinear thermomechanical elliptic physics. The examples create a suite of tutorials which summarize the main concepts for simulating nonlinear thermomechanical elliptic physics.

2.7.1 Example 1: Cantilever Beam

This example shows a three-dimensional nonlinear thermomechanical simulation of a cantilever beam. The beam is fixed on the surface defined by the plane at $X = -5.0$ and loaded on surface defined by the plane at $X = 5.0$.

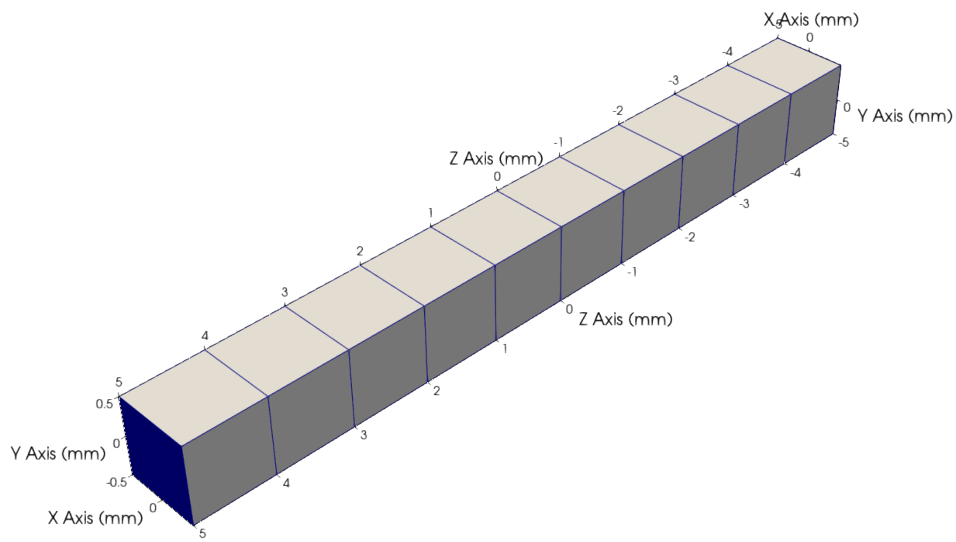

Fig. 2.7.1: The computational domain and the finite element mesh used for the nonlinear thermomechanical analysis. The X, Y, and Z displacements on the surface defined by the plane at $X = -5.0$ are fixed while the temperature is kept at $0.0°$F. A traction force (lbf) with components $(0.0, 0.0, 5e6)$ is applied on the surface defined by the plane at $X = 5.0$ while the temperature is kept at $100.0°$F.

Model Definition

Fig. 2.7.1 shows the computational domain and the finite element mesh used for the nonlinear thermomechanical analysis. The simulation aims to find the temperature and displacement distributions due to the exposure of the cantilever beam to thermal and mechanical loading. The Dirichlet and Neumann boundary conditions for the nonlinear thermomechanical analysis are:

1. Dirichlet boundary conditions
   a. The X, Y, and Z displacements on the surface defined by the plane at $X = -5.0$ are fixed.
   b. The surface defined by the plane at $X = -5.0$ is kept at $0.0°$F.
   c. The surface defined by the plane at $X = 5.0$ is kept at $100.0°$F.
2. Neumann boundary conditions
   a. A traction force (lbf) with components $(0.0, 0.0, 5e6)$ is applied on the surface defined by the plane at $X = 5.0$.

The material modulus of elasticity is set to 29000 ksi and Poisson's ratio is set to 0.29. The material conductivity constant is 112.0 BTU-in$/$(hr-ft^2-$^\circ$F) or 2.017 lbf$/$(s-$^\circ$F). The coefficient of thermal expansion is set to 9.61 μin (in-$^\circ$F) and the reference temperature is set to 0.0°F.

Input Deck

The input deck setup for the nonlinear thermomechanical analysis is:

Listing 2.7.1: Input deck setup for the simulation of the nonlinear thermomechanical elliptic physics.

```
begin service 1
  code analyze
  number_processors 1
end service

begin scenario 1
  physics steady_state_nonlinear_thermomechanics
  service 1
  dimensions 3
  loads 1
  boundary_conditions 1 2 3
  blocks 1
  output 1
  material 1
end scenario

begin output 1
  service 1
  data dispx dispy dispz temperature
  native_service_output true
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name fixed
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type sideset
  location_name fixed
  degree_of_freedom temp
  value 0.
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type sideset
  location_name load
  degree_of_freedom temp
```

(continues on next page)

(continued from previous page)

```
  value 100.
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name load
  value 0 0 5e6
end load

begin block 1
  material 1
  name block_1
end block

begin material 1
  material_model isotropic_thermal_hyperelastic_neohookean
  poissons_ratio 0.29
  youngs_modulus 29000e3
  thermal_expansivity 9.61e-6
  thermal_conductivity 2.017
  reference_temperature 0.0
end material

begin study
  method analysis
end study

begin mesh
  name elliptic_nonlinear_thermomechanical_ex1.exo
end mesh
```

Run Study

To run the nonlinear thermomechanical simulation from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py elliptic_nonlinear_thermomechanical_ex1.i
```

`runmorphorm.py` is the python script with the sequence of instructions to run the Morphorm software and `elliptic_nonlinear_thermomechanical_ex1.i` is the *input deck* for the nonlinear thermomechanical simulation. The simulation should run after pressing the `Enter` key.

Results

The displacement and temperature fields for the cantilever beam are shown in Fig. 2.7.2. and Fig. 2.7.3. The expected displacement and temperature distributions are computed by the nonlinear thermomechanical simulation.

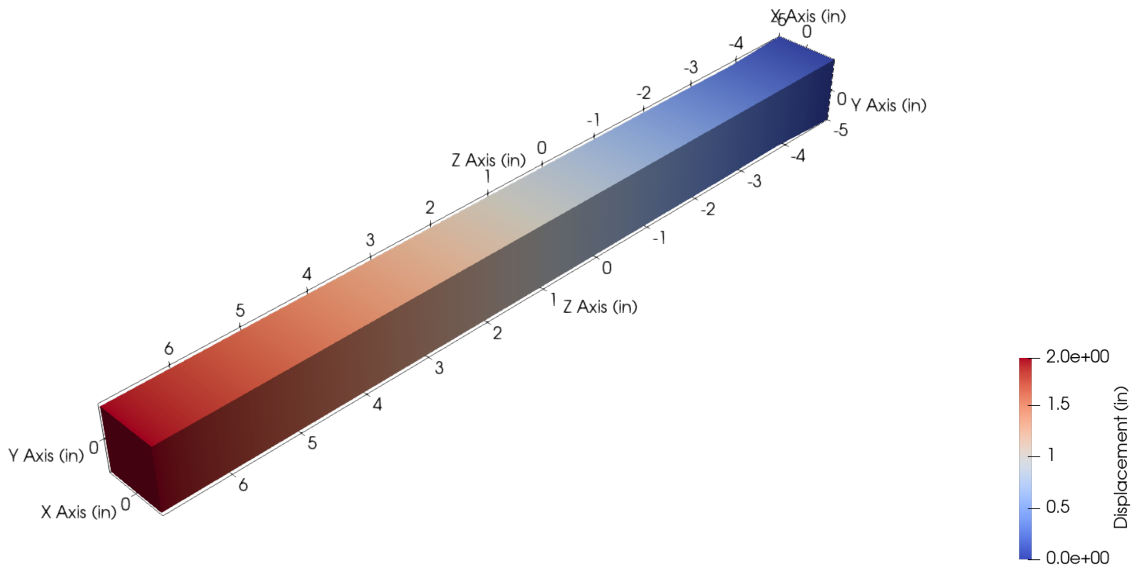

Fig. 2.7.2: The displacement field for the cantilever beam.

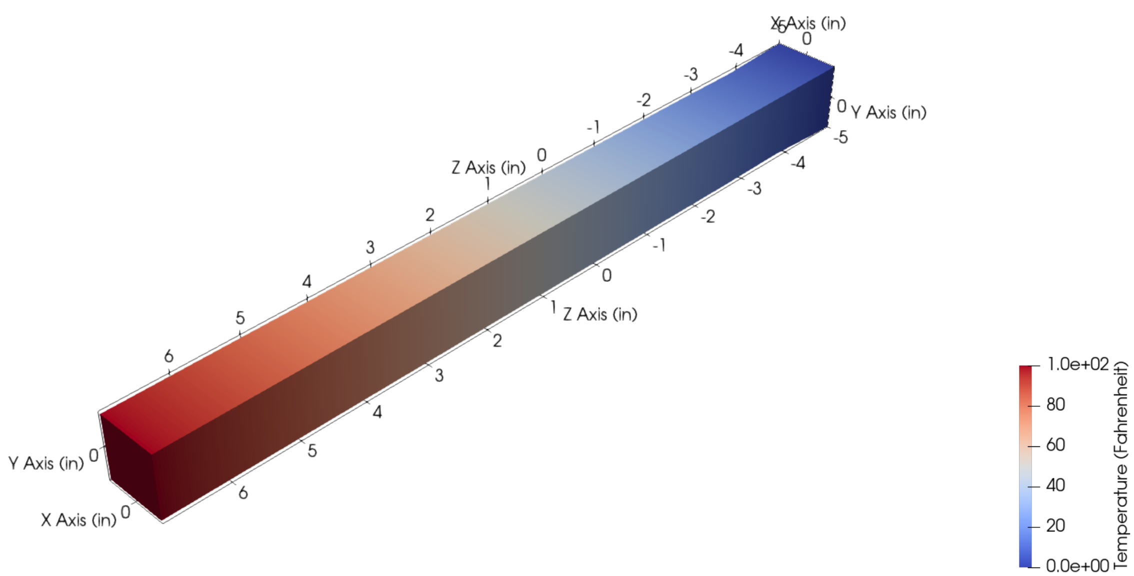

Fig. 2.7.3: The temperature field for the cantilever beam.

CHAPTER

THREE

TOPOLOGY OPTIMIZATION

The Topology Optimization chapter provides a suite of examples describing how to use the Optimize module to set up and run topology optimization problems.

3.1 Description

Topology optimization [Bendsoe and Sigmund] is a mathematical method used to optimize the material layout of a physical system given a set of mathematical objective and constraint functions. Topology optimization is not shape optimization, which is defined in this manuscript as a mathematical method used to optimize the parametric shape parameters that form the geometry. In topology optimization, the optimizer is free to select any material layout within the design space as long as it satisfies the objective and constraint functions. In contrast, shape optimization methods, operate on a restricted set of geometry parameters, i.e., shape parameters, that define the design.

3.2 Mechanical

The following topology optimization examples demonstrate how to use density-based topology optimization to optimize the distribution of a fixed amount of material such that the mechanical design criteria are improved. The examples create a suite of tutorials which summarize the main concepts for solving topology optimization problems modeling mechanical physics and small deformations.

3.2.1 Example 1: Compliance Minimization

This structural topology optimization problem seeks to find a lightweight structure such that the stiffness of the structure is maximized and the mass budget is satisfied. Therefore, the optimizer must balance several conflicting design criteria: stiffness and mass.

Model Definition

Fig. 3.2.1 shows the design space. The loading bracket is made of a linear elastic material. The boundary conditions for the structural topology optimization study are:

1. Dirichlet boundary conditions

   a. The X, Y, and Z displacements (in) on the surface defined by the plane at $X = 0$ are fixed.

   b. The X, Y, and Z displacements on the inner bolthole surfaces (red inner surfaces in Fig. 3.2.1) are fixed.

2. Neumann boundary conditions

   a. A traction force (lbf) with components $(0, -5.0, 0)$ is applied at the pink bolthole surface, see Fig. 3.2.1.

   b. A traction force with components $(0, -5.0, 0)$ is applied at the cyan bolthole surface, see Fig. 3.2.1.

3. Non-optimizable regions

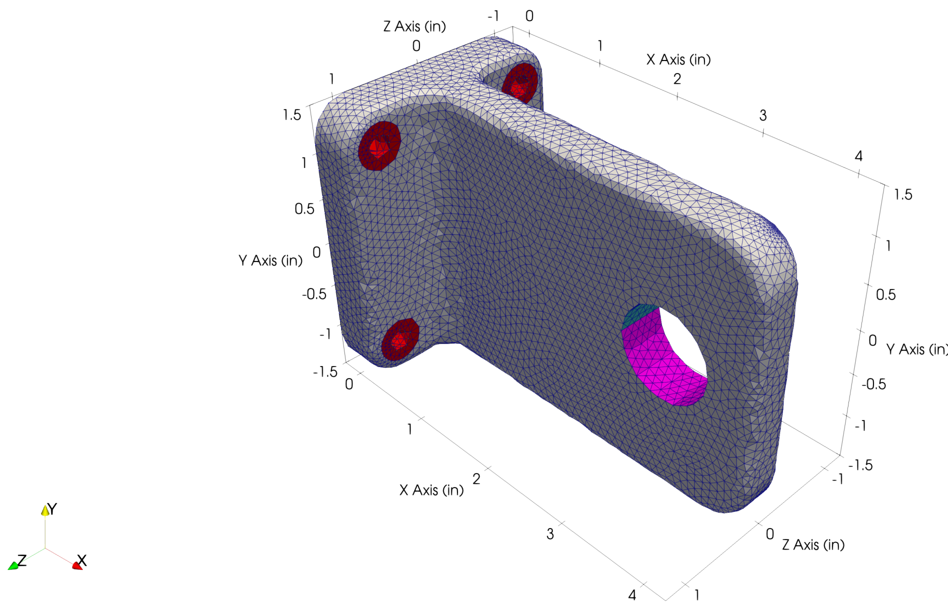

Fig. 3.2.1: The computational domain and the finite element mesh used for the topology optimization study. The X, Y, and Z displacements (:math`textrm{in}`) on the surface defined by the plane at $X = 0$ are fixed. The X, Y, and Z displacements on the inner bolthole surfaces (red surfaces) are fixed. A traction force (lbf) with components $(0, -5.0, 0)$ is applied at the pink bolthole surface. A second traction force with components $(0, -5.0, 0)$ is applied at the cyan bolthole surface.

a. The nodes on the bottom surface red are set to non-optimizable nodes.

The elastic modulus is set to 29007548 psi and Poisson's ratio is set to 0.33.

Formulation

The structural topology optimization problem under consideration seeks to maximize the structural stiffness while satisfying the mass budget. Mathematically, a structural topology optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_z}} \quad & \alpha f(\mathbf{u}(\mathbf{z}),\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0 \\ & h(\mathbf{z}) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\ldots,N_z \end{aligned} \tag{3.2.1}$$

where $\mathbf{u}(\mathbf{z})$ is the displacement vector, which depends on the vector of design variables $\mathbf{z}$ of size N_z. The objective function $f(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is defined by the mechanical compliance criterion:

$$f(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \int_\Omega k_{ij}^e(\mathbf{z}) u_i u_j \, d\Omega, \quad \text{for} \quad i,j = 1,\ldots,d \tag{3.2.2}$$

where d is the spatial dimension, $k_{ij}^e(\mathbf{z})$ is the element stiffness matrix, u_i is the i-th component of the displacement vector. The mechanical compliance criterion is weighted by the scalar $\alpha = 1.0$. $\mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the residual for the linear mechanical elliptic physics. The mass budget constraint $h(\mathbf{z})$ is defined as:

$$h(\mathbf{z}) = \int_\Omega \mathbf{z} \, d\Omega \tag{3.2.3}$$

The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum member size is introduced via the length scale parameter of the Kernel filter [Bourdin] to discourage the optimizer from creating thin structural members. The method of moving asymptotes [Svanberg] is used to solve Eq.3.2.1.

Input Deck

The input deck setup for the structural topology optimization study is:

Listing 3.2.1: Input deck setup for the structural topology optimization study.

```
begin service 1
 code engine
 number_processors 1
 number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type mechanical_compliance
  minimum_ersatz_material_value 1e-9
end criterion

```

(continues on next page)

(continued from previous page)

```
begin criterion 2
  type volume
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1 2
  minimum_ersatz_material_value 1e-9
end scenario

begin objective
  type single_criterion
  criteria 1
  services 2
  scenarios 1
end objective

begin output
  service 2
  output_data true
  data dispx dispy dispz
  native_service_output true
  output_frequency 1
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name ns_1
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name ss_1
  value 0 -5. 0
end load

begin load 2
```

(continues on next page)

(continued from previous page)

```
  type traction
  location_type sideset
  location_name ss_2
  value 0 -5. 0
end load

begin constraint 1
  criterion 2
  scenario 1
  relative_target 0.3
  type less_than
  service 2
end constraint

begin block 1
  material steel
  name block_1
end block

begin block 2
  material steel
  name block_2
end block

begin material steel
  material_model isotropic_linear_elastic
  poissons_ratio 0.33
  youngs_modulus 29007548
end material

begin study
  method topology
  discretization density
  initial_density_value 1.0
  optimization_algorithm oc
  max_iterations 100
  filter_radius_scale 2
  normalize_in_aggregator true
  fixed_block_ids 2
  fixed_exodus_sideset_names ss_1 ss_2
  fixed_exodus_nodeset_names ns_1 ns_2
end study

begin mesh
  name topology_optimization_mech_ex1.exo
end mesh
```

Run Study

To run the structural topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex1.i
```

runmorphorm.py is the python script holding the sequence of instructions to run the Morphorm software and topology_optimization_mech_ex1.i is the *input deck* for the structural topology optimization study. The design study should run after pressing the Enter key.

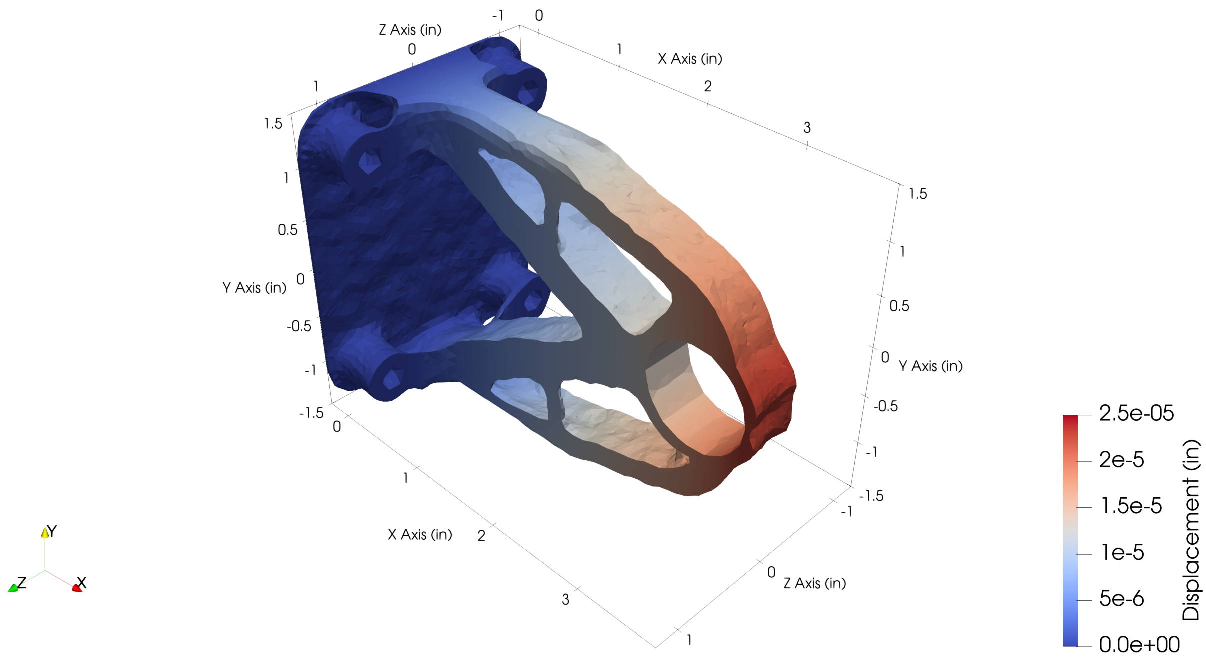

Fig. 3.2.2: Displacement field for the topology optimized bracket.

Results

Fig. 3.2.2 shows the displacement field for the topology optimized bracket. The 70 percent mass reduction goal was achieved while the structural stiffness was maximized. The topology optimizer was successful in finding a material layout that achieved the design goals.

3.2.2 Example 2: Prune & Refine

This structural topology optimization example applies the prune and refine tool to improve the results from *Example 1*. The design goals remain the same, find a lightweight structure such that the structural stiffness is maximized and the mass budget is satisfied.

The prune and refine tool aims to reduce computational cost while providing a mechanism to resolve complex design features. Multiple levels of uniform mesh refinement can be requested to produce smooth and connected design features. Furthermore, void elements, e.g., low stiffness elements, can be pruned to improve simulation predictions in the subsequent topology optimization study. In summary, the prune and refine tool enables:

- pruning of void elements to improve simulation predictions in a subsequent topology optimization study, or
- creating a computational model for a verification study.

Fig. 3.2.3 shows two initial computational domains. The initial computational domain shown on the left pane was used as the starting guess for *Example 1: Compliance Minimization*. The computational domain show on the right pane is the design space after the prune and refine tool was used to prune void elements. One level of uniform mesh refinement

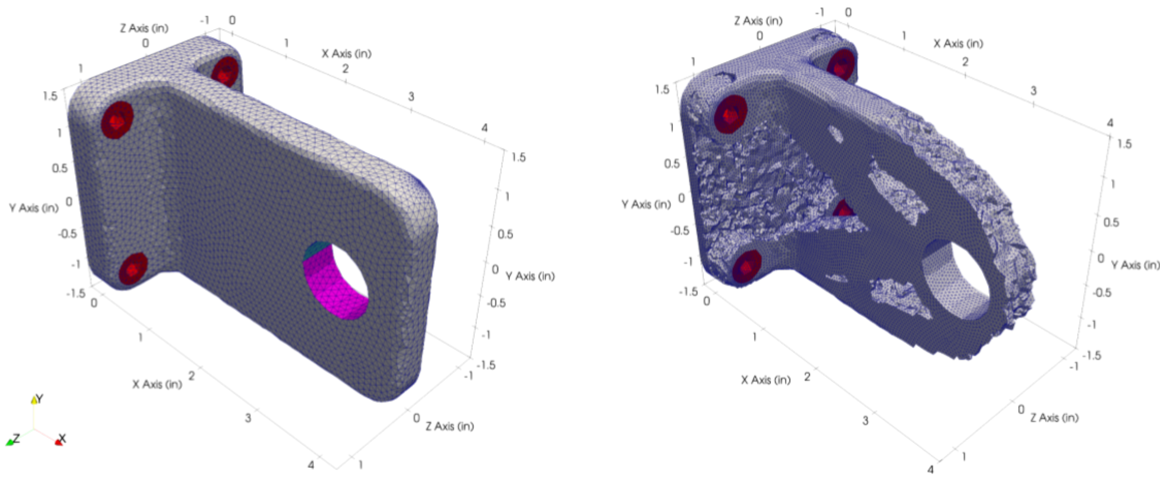

Fig. 3.2.3: The left pane shows the computational domain and the finite element mesh used for the initial *topology optimization study*. The right pane shows the computational domain and the finite element mesh produced with the prune and refine tool. The computational model in the right pane was used for the subsequent structural topology optimization study. The X, Y, and Z displacements (in) on the surface defined by the plane at $X = 0$ are fixed. The X, Y, and Z displacements on the inner bolt hole surfaces are fixed. A traction force (lbf) with components $(0.0, -5.0, 0.0)$ is applied at the pink bolthole surface. A second traction force with components $(0.0, -5.0, 0.0)$ is applied at the cyan bolthole surface.

was applied to allow the optimizer to resolve small design features. The computational domain in the right pane was used as the starting guess for *Example 2: Prune & Refine*. The boundary conditions, elastic material properties, and topology optimization formulation are the same as the ones used for *Example 1: Compliance Minimization*.

Input Deck

The following input deck setup is for a design workflow that combines the prune and refine method with topology optimization (PR-TO). The workflow assumes the existence of a restart file, e.g., engine_output.exo, in the run directory containing a topology field, i.e., density or level set field. Furthermore, two additional input deck blocks must be added to the input deck in *Example 1: Compliance Minimization* to set up the PR-TO workflow. The first block is an additional `service block` to enable the prune and refine application. The second block is the `prune block`, which is used to define the input parameters for the prune and refine application.

Listing 3.2.2: Input deck setup for the design workflow that combines the prune and refine method with topology optimization.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service
```

(continues on next page)

(continued from previous page)

```
begin service 3
  code prune
  number_processors 4
end service

begin criterion 1
  type mechanical_compliance
  minimum_ersatz_material_value 1e-9
end criterion

begin criterion 2
  type volume
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1 2
  minimum_ersatz_material_value 1e-9
end scenario

begin objective
  type single_criterion
  criteria 1
  services 2
  scenarios 1
end objective

begin output
  service 2
  output_data true
  data dispx dispy dispz
  native_service_output true
  output_frequency 1
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name ns_1
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom dispx dispz dispy
```

(continues on next page)

(continued from previous page)

```
  value 0 0 0
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name ss_1
  value 0 -5. 0
end load

begin load 2
  type traction
  location_type sideset
  location_name ss_2
  value 0 -5. 0
end load

begin constraint 1
  criterion 2
  scenario 1
  absolute_target 2.75156890293
  type less_than
  service 2
end constraint

begin block 1
  material steel
  name block_1
end block

begin block 2
  material steel
  name block_2
end block

begin material steel
  material_model isotropic_linear_elastic
  poissons_ratio 0.33
  youngs_modulus 29007548
end material

begin prune
  number_of_refines 1
  number_buffer_layers 2
  initial_guess_mesh engine_output.exo
end prune

begin study
  method prune_and_topology
  discretization density
  initial_density_value 1.0
  optimization_algorithm oc
```

(continues on next page)

(continued from previous page)

```
  max_iterations 100
  filter_radius_scale 2
  normalize_in_aggregator true
  fixed_block_ids 2
  fixed_exodus_sideset_names ss_1 ss_2
  fixed_exodus_nodeset_names ns_1 ns_2
end study

begin mesh
  name topology_optimization_mech_ex2.exo
end mesh
```

The `relative_target` keyword in the `constraint block` was replaced with the `absolute_target` keyword to avoid the need to update the target volume every time the mesh is pruned. The absolute target volume can be calculated by multiplying the original design volume by the relative target volume, e.g., the volume fraction.

Run Study

To run the structural topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex2.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_mech_ex2.i` is the *input deck* for the PR-TO design workflow. The design study should run after pressing the `Enter` key.

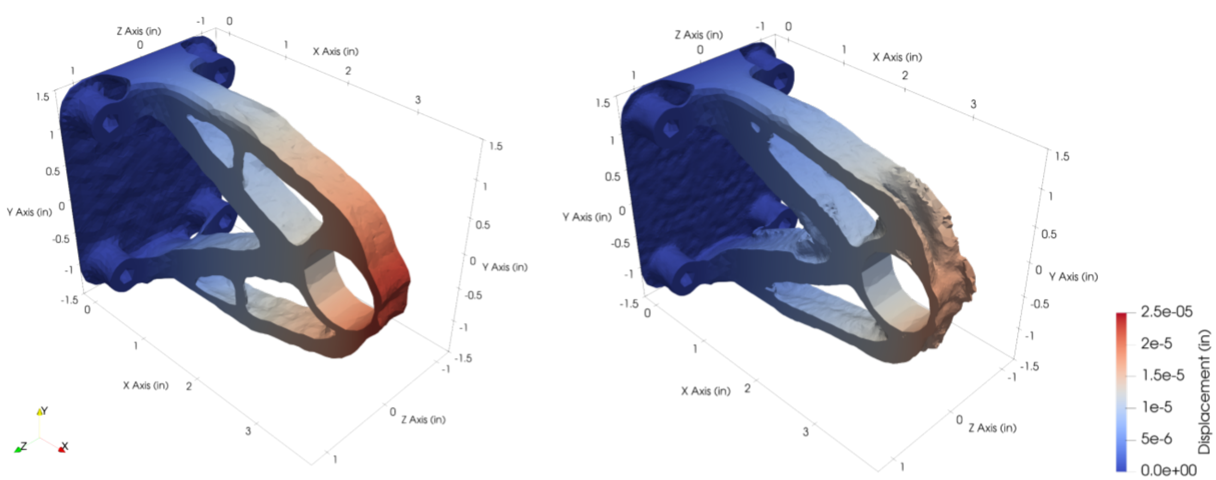

Fig. 3.2.4: Displacement distribution for the optimized brackets. The left pane shows the displacement field for the optimized bracket from *Example 1: Compliance Minimization*. The right pane shows the displacement field for the optimized bracket produced with the PR-TO workflow.

Results

Fig. 3.2.2 shows the displacement fields for the optimized brackets. The optimized bracket produced with the PR-TO workflow exhibits a lower peak displacement than the optimized bracket from *Example 1: Compliance Minimization*. Both solutions meet the 70 percent mass reduction target. The prune and refine approach can be repeated until the desired smoothness is achieved.

3.2.3 Example 3: Local Stress Constraints

This topology optimization example seeks to find a lightweight structure that satisfies a stress limit at every material point. Topology optimization problems with local stress constraints become computationally intractable if the local stress constraints are not modeled correctly. Indeed, at a minimum, the simulation must be called $1+N_g$ times at every major optimization iteration k to compute the gradients for the local stress constraints, where N_g is the number of stress constraints. Each call to the simulation requires solving a large system of equations. A major optimization iteration is defined in this manuscript as the iteration performing the design variable updates. To mitigate the computational burden of topology optimization problems with local stress constraints, an augmented Lagrangian approach is applied to efficiently model the large number of stress constraints.

Model Definition

Fig. 3.2.5 shows the computational domain for the stress constrained topology optimization study, which is made of a linear elastic material. The boundary conditions are:

1. Dirichlet boundary conditions
   a. The X, Y, and Z displacements (m) on the green surface, i.e., plane at $Y = 0$, are fixed, see Fig. 3.2.5.
2. Neumann boundary conditions
   a. A traction force (MPa) with components $(0., -1., 0.)$ is applied at the blue surface shown in Fig. 3.2.5.

The elastic modulus is set to 1.0 GPa and Poisson's ratio is set to 0.3.

Formulation

The topology optimization problem with local stress constraints seeks to find a lightweight bracket such that the stress limit is met at every material point. The optimization problem under consideration can be formulated as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}), \mathbf{z}\right) \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}), \mathbf{z}\right) = 0 \\ & g_j\left(\mathbf{u}(\mathbf{z}), \mathbf{z}\right) \leq 0,\ j = 1, \dots, N_g \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1, \dots, N_z, \end{aligned} \tag{3.2.4}$$

where $N_f = 1$ is the number of objective functions and N_g is the number of nonlinear constraints. The objective function $f_{l=1}$ is defined by the volume criterion:

$$f_{l=1}(\mathbf{z}) = \int_\Omega \mathbf{z}\, d\Omega \tag{3.2.5}$$

where Ω is the design domain and $\mathbf{u}(\mathbf{z})$ is the displacement vector, which depends on the vector of design variables $\mathbf{z}$ of size N_z. The objective function is weighted by the scalar $\alpha_{l=1} = 1.0$. $\mathbf{R}(\mathbf{u}(\mathbf{z}), \mathbf{z})$ is the residual for the linear mechanical physics. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum member size is introduced via the length scale parameter of the Kernel filter. Finally, the method of moving asymptotes is used to solve Eq.3.2.1.

The optimization problem in Eq.3.2.4 becomes computationally intractable if the nonlinear constraints g_j are not properly managed. An augmented Lagrangian method is used instead to efficiently model the large number of nonlinear

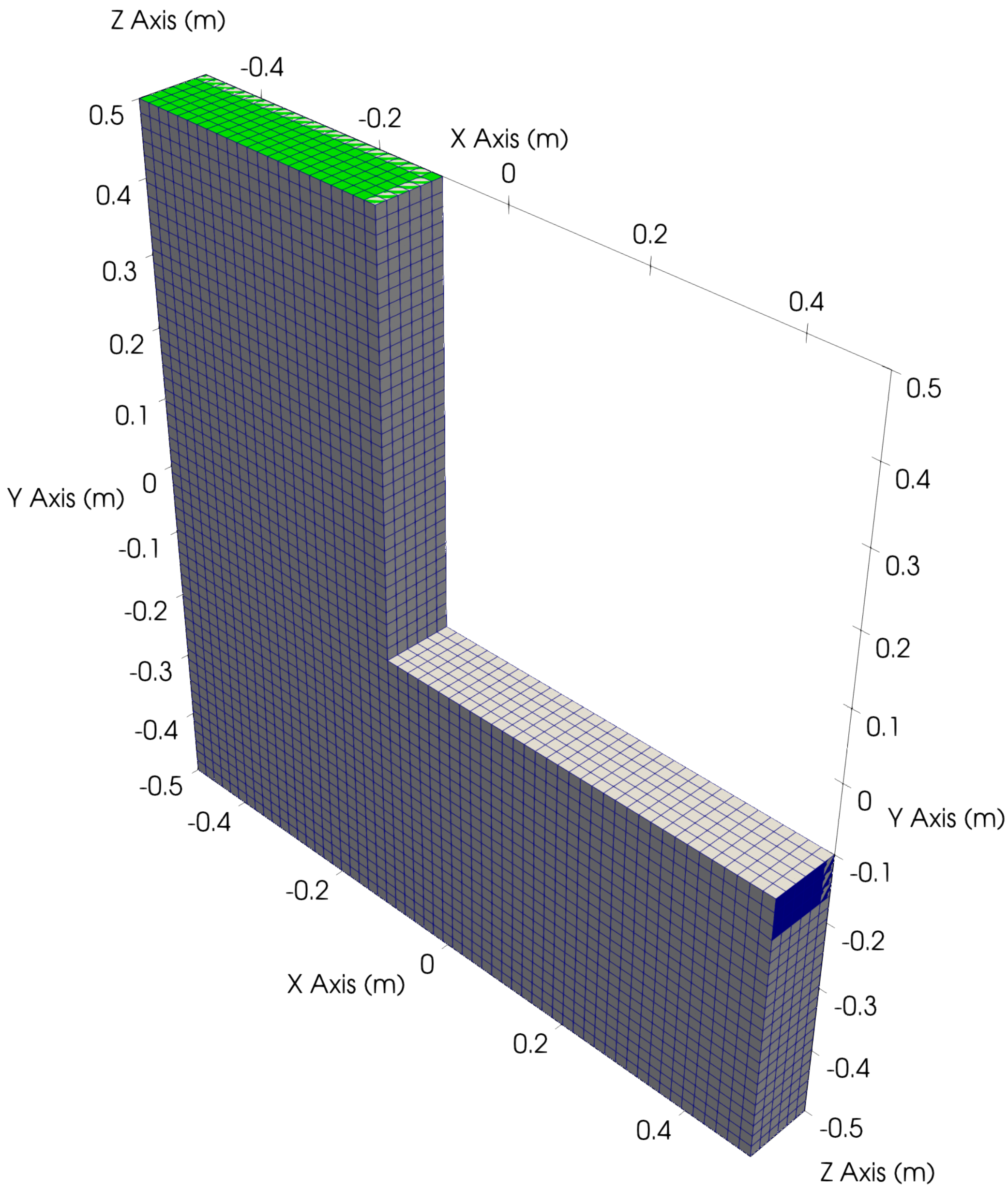

Fig. 3.2.5: The initial computational domain for the topology optimization study. The X, Y, and Z displacements (m) on the green surface, i.e., plane at $Y = 0$, are fixed. A traction force (MPa) with components $(0., -1., 0.)$ is applied at the blue surface.

constraints in Eq.3.2.4 [Giraldo-Londono *et al.*]. At each major optimization iteration k, a sequence of optimization problems is solved to find an optimal and feasible solution $\mathbf{z}$. Mathematically, the topology optimization problem with local stress constraints, Eq.3.2.4, is recast as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) + \frac{\beta}{N_g}\sum_{j=1}^{N_g}\left[\gamma_j^{(k)}\hat{g}_j\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) + \frac{\mu_j^{(k)}}{2}\hat{g}_j\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right)^2\right] \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) = 0 \\ & \underline{z}_i \le z_i \le \bar{z}_i,\ i = 1,\ldots,N_z, \end{aligned} \tag{3.2.6}$$

where $\beta > 0$ and

$$\hat{g}_j\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) = \left(g_j\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right), -\frac{\gamma_j^{(k)}}{\mu_j^{(k)}}\right), \tag{3.2.7}$$

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} + \mu_j^{(k)}\hat{g}_j\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right), \tag{3.2.8}$$

$$\mu_j^{(k+1)} = \left(\alpha\mu_j^{(k)}, \mu_{max}\right),\ \alpha > 1. \tag{3.2.9}$$

The augmented Lagrangian problem in Eq.3.2.6 is solved until the convergence criteria are met. The parameters $\gamma_j^{(k)}$ and $\mu_j^{(k)}$ are the j-th Lagrange multiplier and the j-th penalty, respectively. These two parameters are associated with the j-th nonlinear constraints $g_j^{(k)}$.

Input Deck

The following text snippet is the input deck setup for the topology optimization problem with local stress constraints:

Listing 3.2.3: Input deck setup for the topology optimization problem with local stress constraints.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 10
end service

begin criterion 1
  type local_constraint
  limits 5e6
  local_measures vonmises
  location_name block_1
  initial_penalty 0.1
  minimum_ersatz_material_value 1e-6
```

(continues on next page)

(continued from previous page)

```
end criterion

begin criterion 2
  type volume
  location_name block_1
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1
  boundary_conditions 1
  material 1
  minimum_ersatz_material_value 1e-6
end scenario

begin objective
  type weighted_sum
  criteria 1 2
  services 2 2
  scenarios 1 1
  weights 1e2 1
end objective

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name ss_1
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name ss_2
  value 0 -1e6 0
end load

begin block 1
  material 1
  name block_1
end block

begin material 1
  material_model isotropic_linear_elastic
  youngs_modulus 1e9
  poissons_ratio 0.3
end material

begin output
```

(continues on next page)

(continued from previous page)

```
  service 2
  native_service_output true
  data vonmises dispx dispy dispz
end output

begin study
  method topology
  discretization density
  initial_density_value 1
  filter_type kernel
  filter_radius 1.5

  max_iterations 100
  move_limit 0.1
  output_frequency 1000
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  fixed_exodus_sideset_names ss_2
end study

begin mesh
 name topology_optimization_mech_ex3.exo
end mesh
```

The `update_problem_frequency` keyword in the `begin service 2` block is used to set the rate at which the Lagrange multipliers in Eq.3.2.8 and the penalties in Eq.3.2.9 are updated.

Run Study

To run the topology optimization study with local stress constraints from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex3.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_mech_ex3.i` is the *input deck* for the topology optimization problem with local stress constraints . The design study should run after pressing the `Enter` key.

Results

Fig. 3.2.6 shows the displacement and von Mises stress fields for the topology optimized bracket. The von Mises stress limit is satisfied at each material point. Furthermore, the total mass of the optimized bracket is 72 percent less than the total mass of the baseline bracket. The topology optimizer required 100 iterations to find an optimized bracket that met the von Mises stress limit at every material point. A total of $13,000$ nonlinear constraints were modeled in the topology optimization study.

3.2.4 Example 4: Center of Gravity

This topology optimization example seeks to find a stiff and lightweight bracket such that the center of gravity is located at a desired location. The optimizer must balance several conflicting design criteria in this example which include the mass budget, the structural stiffness, and the center of gravity.

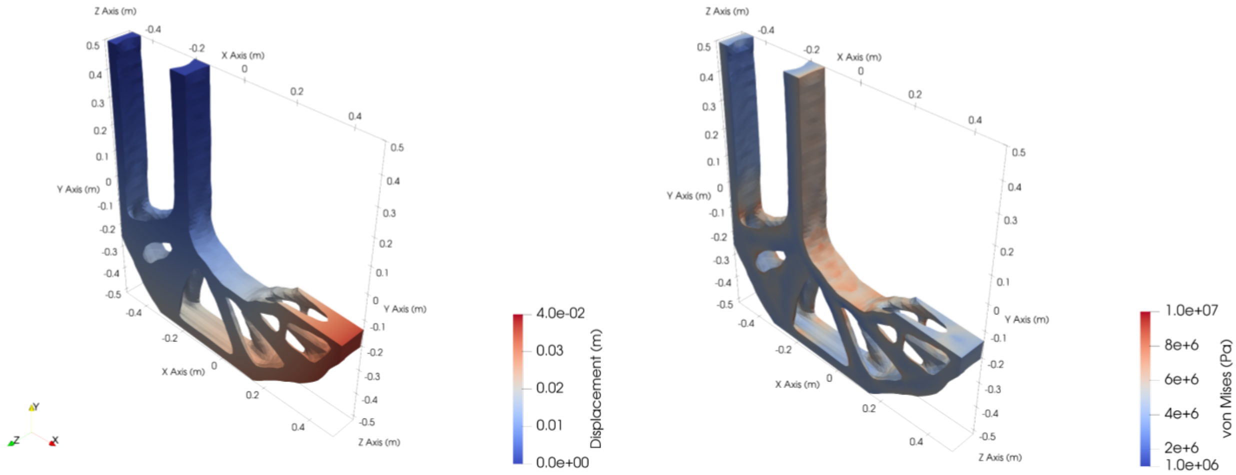

Fig. 3.2.6: The displacement (left pane) and von Mises stress (right pane) fields for the topology optimized bracket.

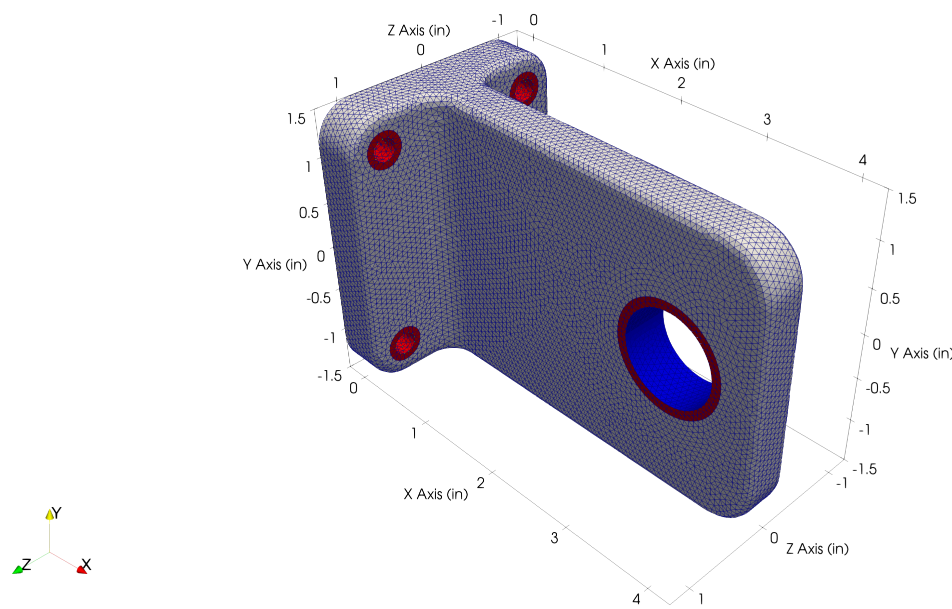

Fig. 3.2.7: The computational domain and the finite element mesh for the topology optimization study. The X, Y, and Z displacements ($textrmin$) on the inner bolthole red surfaces are fixed. A traction force (lbf) is applied at the blue surface. The red subdomains (regions) are set as frozen, i.e., non-optimizable, regions. Thus, the optimizer is not allow to remove or add material in these regions.

Model Definition

Fig. 3.2.7 shows the initial computational domain, which is made of a linear elastic material. The boundary conditions for the topology optimization study are:

1. Dirichlet boundary conditions

    a. The X, Y, and Z displacements ($textrmin$) on the inner bolthole red surfaces are fixed, see Fig. 3.2.7.

2. Neumann boundary conditions

    a. A traction force (lbf) is applied at the blue surface, see Fig. 3.2.7.

3. Non-optimizable regions

    a. The red subdomains are set to non-optimizable regions.

The elastic modulus is set to 145.0 ksi and Poisson's ratio is set to 0.3.

Formulation

The topology optimization problem seeks to find a stiff and lightweight bracket such that the center of gravity is situated at the desired location. The topoplogy optimization problem under consideration is formulated as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) = 0 \\ & h\left(\mathbf{z}\right) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i, \; i = 1,\ldots,N_z, \end{aligned} \tag{3.2.10}$$

where N_f is the number of objective functions. Each objective function f_l is weighted by a positive scalar $\alpha_l > 0$. $\mathbf{u}(\mathbf{z})$ is the displacement vector, which depends on the vector of design variables $\mathbf{z}$ of size N_z. $\mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the residual vector for the linear mechanical physics and $h(\mathbf{z})$ is the mass budget constraint. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum member size is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.2.10.

The center of gravity misfit criterion is defined as:

$$f_{l=1} = \frac{CG_n(\mathbf{z}) - \widehat{CG}_n}{X_n^{extent}}, \tag{3.2.11}$$

where $CG_i(\mathbf{z})$ is the n-th component of the center of gravity and $\widehat{CG}_n$ is the n-th component of the desired center of gravity. X_n^{extent} is the length of the part in the corresponding coordinate axis. Mathematically, The center of gravity is defined as:

$$CG_n(\mathbf{z}) = \frac{\int_\Omega \mathbf{z}\rho r_n \, d\Omega}{m(\mathbf{z})}, \tag{3.2.12}$$

where ρ is the material density, Ω is the computational domain. The mass of the bracket $m(\mathbf{z})$ is computed using:

$$m(\mathbf{z}) = \int_\Omega \mathbf{z}\rho \, d\Omega, \tag{3.2.13}$$

where r_n is the n-th component of the position vector $\mathbf{r}$.

The mechanical compliance criterion is the second objective function considered in Eq.3.2.10 to maximize the stiffness of the bracket. The mechanical compliance criterion is defined as:

$$f_{l=2}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \int_\Omega k_{pq}(\mathbf{z}) u_p u_q \, d\Omega \tag{3.2.14}$$

where k_{pq} is the element stiffness matrix. The p-th and q-th displacements are given by u_p and u_q, respectively. The indices $p, q = 1, \ldots, d$, with $d \in \{1, 2, 3\}$, denote the spatial dimension.

Input Deck

The following is the input deck setup for the topology optimization study in Eq.3.2.10:

Listing 3.2.4: Input deck setup for the topology optimization study with a center of gravity misfit criterion.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin service 3
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type mechanical_compliance
  minimum_ersatz_material_value 1e-6
end criterion

begin criterion 2
  type volume
end criterion

begin criterion 3
  type mass_properties
  cgx 1.8 weight 1
  cgy 0.5 weight 1
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1
  boundary_conditions 1
  material 1
  minimum_ersatz_material_value 1e-6
end scenario

begin objective
  type weighted_sum
  criteria 1 3
  services 2 2
```

(continues on next page)

(continued from previous page)

```
  scenarios 1 1
  weights 1 1e3
end objective

begin constraint 1
  criterion 2
  relative_target 0.4
  type less_than
  service 2
  scenario 1
end constraint

begin output
  service 2
  output_data true
  output_frequency 10
  native_service_output true
  data dispx dispy dispz vonmises
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name ns_1
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name ss_2
  value 0 -14.5 0
end load

begin block 1
  material 1
end block

begin block 2
  material 1
end block

begin material 1
  material_model isotropic_linear_elastic
  poissons_ratio 0.3
  youngs_modulus 145e3
end material

begin study
  method topology
  discretization density
```

(continues on next page)

(continued from previous page)

```
  max_iterations 200
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  filter_type kernel
  filter_radius_scale 2.0
  initial_density_value 1.0
  normalize_in_aggregator false

  projection_type tanh
  filter_projection_start_iteration 100
  filter_projection_update_interval 5
  filter_use_additive_continuation true
  filter_heaviside_min 1
  filter_heaviside_max 100
  filter_heaviside_update 2

  fixed_block_ids 2
end study

begin mesh
  name topology_optimization_mech_ex4.exo
end mesh
```

The kernel filter is combined with the hyperbolic tangent projection filter, i.e., `projection_type tanh`, in the topology optimization study to steer the optimizer towards a clear "$0 - 1$" solution.

Run Study

To run the topology optimization study with a center of gravity criterion, i.e., Eq.3.2.10, from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex4.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_mech_ex4.i` is the *input deck* for the topology optimization study. The design study should run after pressing the `Enter` key.

Results

Fig. 3.2.8 shows the displacement and von Mises stress fields for the topology optimized bracket. The von Mises stress limit is satisfied at each material point. The mass of the optimized bracket was reduced by 60 percent in comparison to the baseline bracket Fig. 3.2.7. The optimized X and Y center of gravity components were 1.7992 and 0.49897, respectively. Table 3.2.1 shows the optimized values for the center of gravity components and the percent errors. The optimizer was successful in finding a lightweight and stiff bracket that met the desired center of gravity requirements.

Table 3.2.1: The target, result, and percent error for each component of the center of gravity.

CG Component	Target (in)	Result (in)	Percent Error
X	1.8	1.7992	-0.044
Y	0.5	0.4990	-0.200

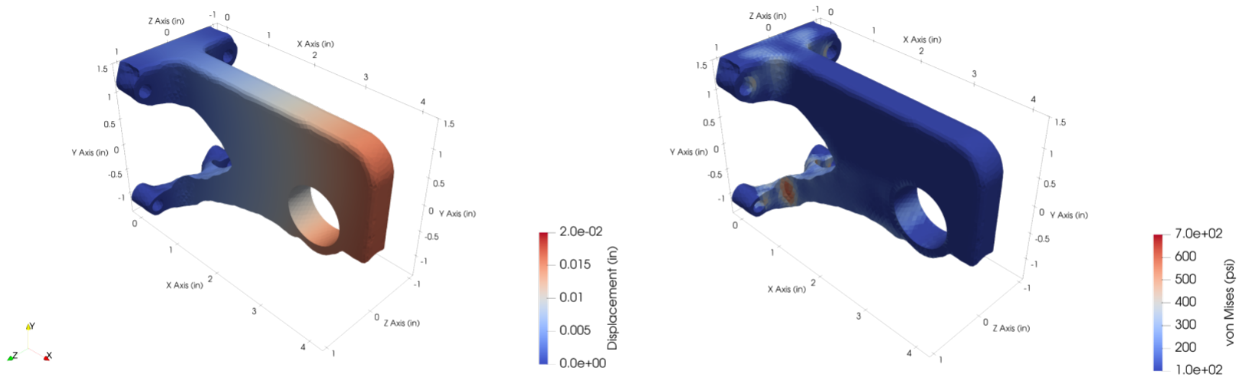

Fig. 3.2.8: The displacement (left pane) and von Mises stress (right pane) fields for the topology optimized bracket.

3.2.5 Example 5: Inverse Criterion

This topology optimization problem seeks to find a lightweight and stiff structure capable of matching a desired deformation profile. The topology optimizer must balance several conflicting design criteria which include a desired deformation profile and mass budget together with stiffness of the structure.

Model Definition

Fig. 3.2.9 shows the computational domain, which is made of a linear elastic material, for the topology optimization study. The boundary conditions for the design study are:

1. Dirichlet boundary conditions
   a. The X, Y, and Z displacements (m) on the green surface defined by the plane at $Y = 2.5$ are fixed.
   b. The X, Y, and Z displacements along the pink edge are fixed.
   c. The X displacements on the red surface defined by the plane at $X = 0.0$ are fixed.
2. External forces
   a. Concentrated forces (N) are applied at the nodes located within the top yellow and blue surfaces. The force components are set to $(0.0, 0.0, -3475.0)$ and $(0.0, 0.0, -6200)$ for the nodes located within the yellow and blue surfaces, respectively.
3. Non-optimizable regions
   a. The nodes located within the yellow and blue surfaces are set to non-optimizable nodes.
   b. The nodes located at the bottom surface are set to non-optimizable nodes.

The elastic modulus is set to 325.0 GPa and Poisson's ratio is set to 0.33.

Formulation

The topology optimization problem with seeks to find a lightweight and stiff structure capable of matching the desired deformation profile for the mission. The optimization problem under consideration is formulated as an inverse problem:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & f\left(\mathbf{u}(\mathbf{z})\right) \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) = 0 \\ & h_1\left(\mathbf{z}\right) \leq 0 \\ & h_2\left(\mathbf{u}(\mathbf{z}),\mathbf{z}\right) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\ldots,N_z, \end{aligned} \tag{3.2.15}$$

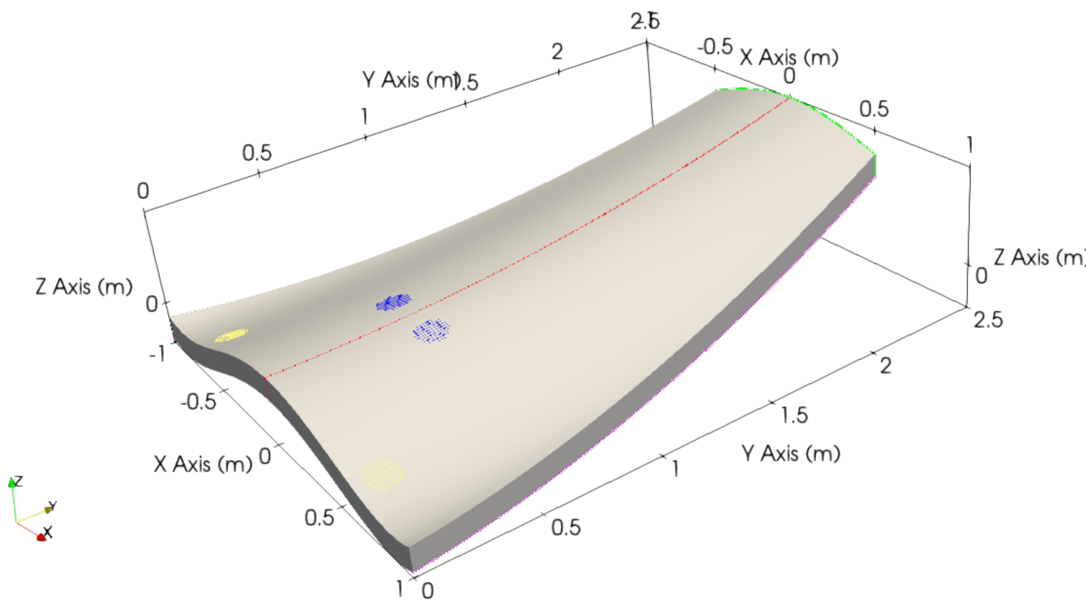

Fig. 3.2.9: The computational domain for the topology optimization study. The X, Y, and Z displacements (m) on the green surface defined by the plane at $Y = 2.5$ are fixed. The X, Y, and Z displacements along the pink edge are fixed. The X displacements on the red surface defined by the plane at $X = 0.0$ are fixed. Concentrated forces are applied at the nodes located on the top yellow and blue surfaces. Furthermore, the nodes on the yellow and blue surfaces are set to non-optimizable nodes. Therefore, material cannot be added nor removed at these nodal locations. Likewise, the nodes on the bottom surface are set to non-optimizable nodes.

where $f(\mathbf{u}(\mathbf{z}))$ is the objective function. The displacement vector $\mathbf{u}(\mathbf{z})$ depends on the vector of design variables $\mathbf{z}$ of size N_z. $\mathbf{R}(\mathbf{u}(\mathbf{z}), \mathbf{z})$ is the residual vector for the linear mechanical physics. The constraints $h_1(\mathbf{z})$ and $h_2(\mathbf{u}(\mathbf{z}), \mathbf{z})$ denote the mass and stiffness constraints, respectively. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum member size is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.2.15.

The objective function $f(\mathbf{u}(\mathbf{z}))$ is defined by the following inverse criterion:

$$f_{\text{inverse}}(\mathbf{u}(\mathbf{z})) = \exp\left(f_{\text{misfit}}(\mathbf{u}(\mathbf{z}))\right), \tag{3.2.16}$$

where the exponential data transformation $\exp(\cdot)$ is applied to mitigate high-frequency oscillations in the displacement misfit criterion $f_{\text{misfit}}(\mathbf{u}(\mathbf{z}))$ term. The displacement misfit criterion is defined as:

$$f_{\text{misfit}}(\mathbf{u}(\mathbf{z})) = \frac{\beta}{2}\|u_p(\mathbf{z}) - \hat{u}_p\|^2, \tag{3.2.17}$$

where $\beta > 0$ is a positive scalar and $p = 1, \ldots, N_p$, with N_p defined as the number of target displacements. The volume criterion is defined as:

$$f_{\text{volume}}(\mathbf{z}) = \int_\Omega \mathbf{z}\, d\Omega, \tag{3.2.18}$$

and the stiffness criterion is defined as:

$$f_{\text{stiff}}(\mathbf{u}(\mathbf{z}), \mathbf{z}) = \int_\Omega k_{ij}(\mathbf{z}) u_i u_j\, d\Omega \tag{3.2.19}$$

where k_{ij} is the element stiffness matrix. The i-th and j-th displacements are given by u_i and u_j, respectively. The indices $i, j = 1, \ldots, d$, with $d \in \{1, 2, 3\}$ specifying the spatial dimension.

Input Deck

The following text snippet is the input deck setup for the inverse topology optimization study defined in Eq.3.2.15:

Listing 3.2.5: Input deck setup for the inverse topology optimization study with a displacement misfit criterion.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type mechanical_compliance
end criterion

begin criterion 2
  type inverse
  criterion_ids 3
  criterion_weights 1
```

(continues on next page)

(continued from previous page)

```
  target_data 3d_pwr_v1_1_1D_surf.csv
  degree_of_freedom dispz
end criterion

begin criterion 3
  type state_misfit
  transform exponential
end criterion

begin criterion 4
  type volume
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1 2 3
  linear_solver 1
  blocks 1
  output 1
  material_penalty_exponent 4.0
end scenario

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 2
  services 2
  scenarios 1
end objective

begin output 1
  service 2
  native_service_output true
  data dispx dispy dispz vonmises
  output_frequency 20
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
```

(continues on next page)

(continued from previous page)

```
  type fixed_value
  location_type nodeset
  location_name symm_plane
  degree_of_freedom dispx
  value 0.
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
  location_name le
  degree_of_freedom dispz
  value 0.
end boundary_condition

begin load 1
  type concentrated_load
  location_type nodeset
  location_name actuator1
  direction z
  value -6200
end load

begin load 2
  type concentrated_load
  location_type nodeset
  location_name actuator2
  direction z
  value -3475
end load

begin constraint 1
  criterion 9
  relative_target 0.25
  type less_than
  service 1
  scenario 1
end constraint

begin constraint 2
  criterion 1
  absolute_target 3e5
  type less_than
  service 2
  scenario 1
end constraint

begin block 1
  material psudoti
  name design_area
end block
```

(continues on next page)

(continued from previous page)

```
begin material psudoti
  material_model isotropic_linear_elastic
  poissons_ratio .33
  youngs_modulus 325e9
end material

begin study
  method topology
  discretization density
  initialize_method uniform
  initial_density_value 1.0

  max_iterations 200
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 2.
  projection_type tanh
  filter_projection_start_iteration 100
  filter_projection_update_interval 5
  filter_use_additive_continuation true
  filter_heaviside_min 1
  filter_heaviside_max 100
  filter_heaviside_update 2

  normalize_in_aggregator false
  fixed_exodus_sideset_names lower_surf_ss
  fixed_exodus_nodeset_names actuator1 actuator2
end study

begin mesh
  name topology_optimization_mech_ex5.exo
end mesh
```

The kernel filter is combined with the hyperbolic tangent projection filter, i.e., `projection_type tanh`, in the topology optimization study to steer the optimizer towards a "$0-1$" design solution.

The prune and refine application is used in *Example 5* to prune void elements from the computational model shown in Fig. 3.2.9. The material layout solution to Eq.3.2.15 is used by the prune and refine application to decide which elements will be kept (solid elements) and which elements will be pruned (void elements). The prune and refine application will create a new computational model that can be used for:

- rerunning the topology optimization study, or
- running a verification simulation to verify the accuracy of the results.

The following text snippet is the input deck setup for the prune and refine application:

Listing 3.2.6: Input deck setup for the prune and refine application.

```
begin service 1
  code prune
  number_processors 8
end service
```

(continues on next page)

(continued from previous page)

```
begin prune
  prune_threshold 0.35
  number_of_refines 1
  spatial_dimensions 3
  number_buffer_layers 2
  initial_guess_field_name Topology
  initial_guess_mesh iteration200.exo
end prune

begin study
  method prune
end study

begin mesh
  name topology_optimization_mech_ex5.exo
end mesh
```

Please consult the `User Manual` to learn how to properly setup the prune and refine application.

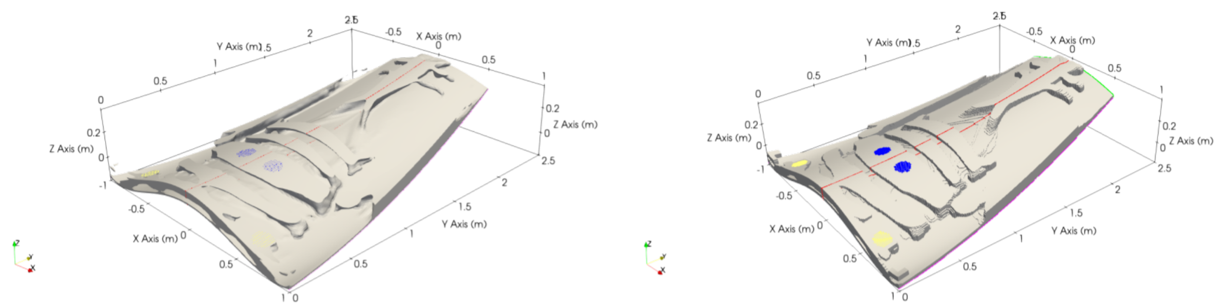

Fig. 3.2.10: The left pane shows the optimized design computed by solving the initial topology optimization study. The right pane shows the design solution produced with the prune and refine application. The output from the prune and refine step was used to run a force calibration problem. The calibrated forces were subsequently used to run a verification simulation with the the pruned and refined computational model. The boundary conditions used for the topology optimization study were also used for the verification simulation. The X, Y, and Z displacements on the green surface defined by the plane at $Y = 2.5$ are fixed. The X, Y, and Z displacements along the pink edge are fixed. The X displacement on the red surface defined by the plane at $X = 0.0$ are fixed. Concentrated forces were applied at the nodes located on the yellow and blue surfaces. The nodes on the yellow and blue surfaces were also set to non-optimizable nodes. Therefore, material cannot be added nor removed at these nodal locations. Likewise, the nodes on the bottom surface were set to non-optimizable nodes.

The design solution produced by the prune and refine application is shown in Fig. 3.2.10. The pruned and refiend design was used to run a force calibration problem. The calibrated forces were subsequently used to run a verification simulation with the pruned and refined computational model. A verification simulation can be easliy run by changing the `method` keyword in the `study block` from `topology` to `analysis`.

Run Study

To run the inverse topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex5.i
```

runmorphorm.py is the python script holding the sequence of instructions to run the Morphorm software and topology_optimization_mech_ex5.i is the *input deck* for the inverse topology optimization study defined in Eq.3.2.15. The design study should run after pressing the Enter key.

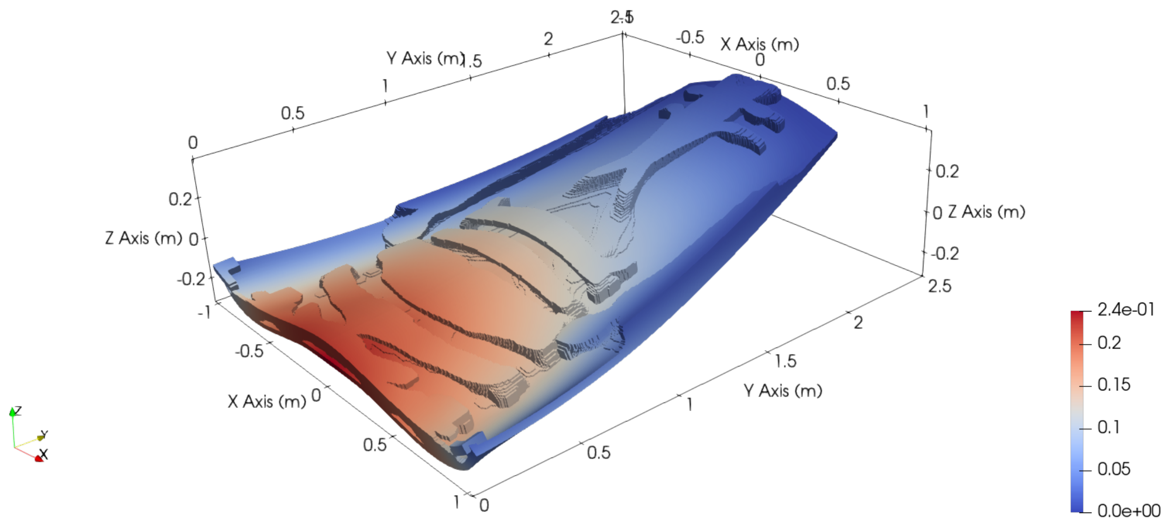

Fig. 3.2.11: The displacement field for the topology optimized structure. The displacement field is computed using the pruned and refined model shown in Fig. 3.2.10.

Results

Fig. 3.2.11 shows the displacement field for the pruned and refined structure. The concentrated forces were recalibrated to match the desired deformation profile. The pruned and refined computational model was used to run the calibation problem. The calibration results are shown in Table 3.2.2, where F_z denotes the concentrated force applied in the Z direction. The calibration results show an increase in magnitude of the actuating forces needed to achieve the desired deformation profile. These results highlight the importance of verifying the topology optimization results when density-based topology optimization methods are used since the density-penalized displacements can be misleading. The low-stiffness elements can adversely impact the accuracy of the results. The mass of the topology optimized structured was 47 percent less than the mass of the baseline design. Furthermore, the misfit error between the computed deformation profile and the targeted deformation profile was approximately 2.0 percent after calibration.

Table 3.2.2: The calibrated concentrated forces. The percent increase is computed using the original and calibrated concentrated forces.

Location	F_x (N)	F_y (N)	F_z (N)	Percent Increase in F_z
Blue Surface	0.0	0.0	-22800	268.0
Yellow Surface	0.0	0.0	-21600	522.0

3.2.6 Example 6: Multi-Level Optimization

This multi-level topology optimization example seeks to find a lightweight structure and the actuating forces needed to match a desired deformation profile. The optimizer must find the actuating forces while simultaneously balancing several conflicting design criteria: deformation profile misfit, structural mass, and structural stiffness. The solution to the multi-level topology optimization problem will be the material layout that meets the design goals plus the actuating forces to achieve the targeted deformation profile.

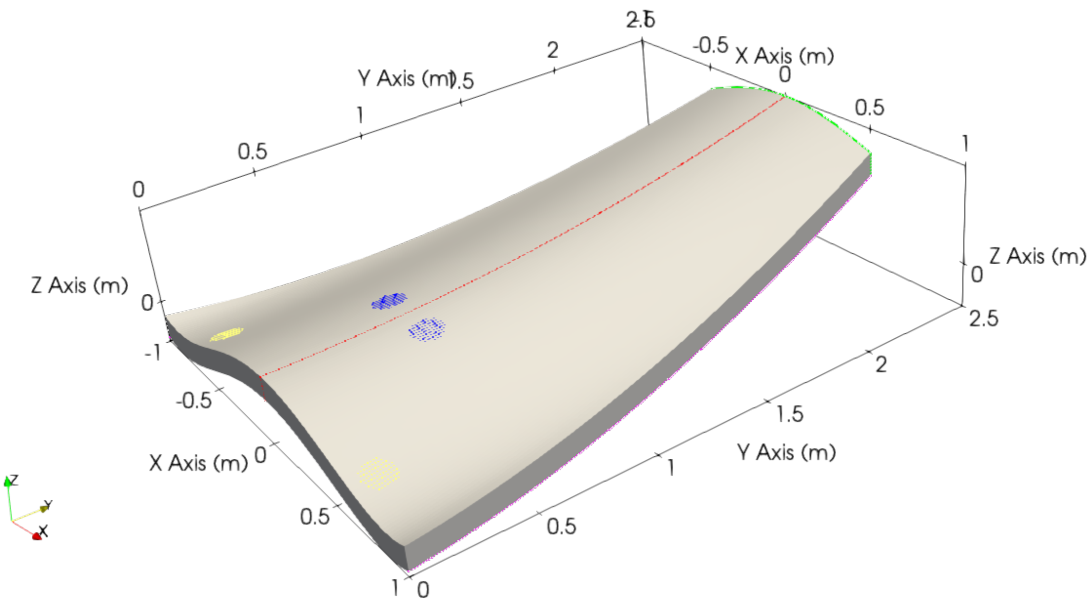

Fig. 3.2.12: The computational domain for the multi-level topology optimization study. The X, Y, and Z displacements (m) on the green surface defined by the plane at $Y = 2.5$ are fixed. The X, Y, and Z displacements along the pink edge are fixed. The X displacements on the red surface defined by the plane at $X = 0.0$ are fixed. Actuating forces (N) are applied at the nodes located on the yellow and blue surfaces. The magnitudes of the forces will be determined through optimization. The nodes on the yellow and blue surfaces are set to non-optimizable nodes. Therefore, material cannot be added nor removed at these nodal locations. Likewise, the nodes on the bottom surface are set to non-optimizable nodes.

Model Definition

Fig. 3.2.12 shows the initial computational domain, which is made of a linear elastic material, for the multi-level topology optimization study. The boundary conditions are:

1. Dirichlet boundary conditions

   a. The X, Y, and Z displacements (m) on the green surface defined by the plane at $Y = 2.5$ are fixed.

   b. The X, Y, and Z displacements along the pink edge are fixed.

   c. The X displacements on the red surface defined by the plane at $X = 0.0$ are fixed.

2. External forces

   a. Actuating forces in the form of concentrated forces (N) are applied at the on the yellow and blue surfaces shown in Fig. 3.2.12. The force components are set to $(0.0, 0.0, -3475.0)$ and $(0.0, 0.0, -6200)$ for the nodes located on the yellow and blue surfaces, respectively.

3. Non-optimizable regions

   a. The nodes on the yellow and blue surfaces are set to non-optimizable nodes.

   b. The nodes on the bottom surface are set to non-optimizable nodes.

The elastic modulus is set to 325.0 GPa and Poisson's ratio is set to 0.33.

Formulation

The multi-level topology optimization problem seeks to find a lightweight structure and the actuating forces needed to match the desired deformation profile. The optimization problem under consideration can be formulated as:

$$
\begin{aligned}
\min_{\mathbf{p}\in\mathbb{R}^{N_\mathbf{p}},\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{M=1}^{N_F} \alpha_M F_M\left(\mathbf{p},\mathbf{z}\right) \\
\text{s.t.} \quad & \underline{p}_i \leq p_i \leq \bar{p}_i \text{ for } i=1,\dots,N_\mathbf{p} \\
& \underline{z}_j \leq z_j \leq \bar{z}_j \text{ for } j=1,\dots,N_\mathbf{z} \\
& H_I\left(\mathbf{p},\mathbf{z}\right) \leq 0 \text{ for } I=1,\dots,N_H \\
& \mathbf{z} \in \arg\min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \{ \sum_{N=1}^{N_f} \beta_N f_N\left(\mathbf{p},\mathbf{z}\right) : \mathbf{R}\left(\mathbf{p},\mathbf{z}\right)=0,\ h_J\left(\mathbf{p},\mathbf{z}\right) \leq 0 \text{ for } J=1,\dots,N_h \}
\end{aligned}
\tag{3.2.20}
$$

where $F_M(\mathbf{p},\mathbf{z})$ and $f_N(\mathbf{p},\mathbf{z})$ are the outer- and inner-level objective functions, which are weighted by scalars $\alpha_M > 0$ and β_N. The outer- and inner-level objective functions depend on the displacement vector $\mathbf{u}(\mathbf{z})$, which implicitly depend on the vectors of design variables $\mathbf{p}$ (force parameters) and $\mathbf{z}$ (material layout parameterization) of size $N_\mathbf{p}$ and $N_\mathbf{z}$, respectively. $\mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the residual vector for the linear mechanical physics. The outer- and inner-level constraints are $H_I(\mathbf{p},\mathbf{z})$ and $h_J\left(\mathbf{p},\mathbf{z}\right)$, respectively. The inner and outer bounds for the i-th outer-level design variable p_i are $\underline{p}_i$ and $\bar{p}_i$, respectively. Likewise, the inner and outer bounds for the i-th inner-level design variable z_i are $\underline{z}_i$ and $\bar{z}_i$, respectively. Finally, a minimum member size is introduced for the inner-level design variables via the length scale parameter of the Kernel filter.

A multi-level optimization method is used to solve Eq.3.2.20. A surrogate-based global optimizer combined with a material layout optimizer is employed to find the actuating forces (outer-level design variables) and the material layout (inner-level design variables). At a major outer-level optimization iteration, the best actuating forces are passed to the inner-level gradient-based optimizer to find the material layout. The method of moving asymptotes is used to solve the inner level optimization problem in Eq.3.2.20.

Design Criteria

The outer- and inner-level objective functions are set to the inverse deformation profile criterion, which is defined as:

$$F_{\text{inverse}}\left(\mathbf{u}(\mathbf{z})\right) = f_{\text{inverse}}\left(\mathbf{u}(\mathbf{z})\right) = \exp\left(\hat{f}_{\text{misfit}}\left(\mathbf{u}(\mathbf{z})\right)\right) \tag{3.2.21}$$

where the exponential data transformation $\exp\left(\cdot\right)$ is applied to mitigate high-frequency oscillations in the displacement misfit criterion $\hat{f}_{\text{misfit}}\left(\mathbf{u}(\mathbf{z})\right)$ term. The displacement misfit criterion is defined as:

$$\hat{f}_{\text{misfit}}\left(\mathbf{u}(\mathbf{z})\right) = \frac{\gamma}{2}\|u_p(\mathbf{z}) - \hat{u}_p\|^2 \tag{3.2.22}$$

where $\gamma > 0$ is a positive scalar and $p = 1,\dots,d$, with $d \in \{1,2,3\}$ specifying the spatial dimension. The outer- and inner-level optimization problems seek to find the combination of actuating forces and material layout that minimizes Eq.3.2.21.

Additional design constraints are enforced in the inner-level optimization problem in the form of a volume budget and structural stiffness constraints. The volume criterion is defined as:

$$h_{J=1}(\mathbf{z}) = \int_\Omega \mathbf{z}\, d\Omega \tag{3.2.23}$$

and the stiffness criterion is defined as:

$$h_{J=2}(\mathbf{p},\mathbf{z}) = \int_\Omega k_{pq}(\mathbf{z}) u_p(\mathbf{p}) u_q(\mathbf{p})\, d\Omega \tag{3.2.24}$$

where $k_{pq}(\mathbf{z})$ is the material stiffness matrix and $p,q = 1,\dots,d$, with $d \in \{1,2,3\}$ denoting the spatial dimension. The p-th and q-th displacements are given by u_p and u_q, respectively.

Input Deck

Two input decks are needed to solve the multi-level structural topology optimization problem defined in Eq.3.2.20. The first input deck is for the actuating force optimization problem and the second input deck is for the material layout optimizatation problem.

Outer Optimizer

The following input deck setup is for the outer-level optimization problem:

Listing 3.2.7: Input deck setup for the outer-level optimization problem.

```
begin variables 1
  numvars 2
  type continuous_design
  initial_points  6e3   4e3
  lower_bounds    1e3   1e3
  upper_bounds    5e4   5e4
  descriptors    load1 load2
end variables

begin study
  method surrogate_based_global_optimization
  variables 1
  num_samples 8
  num_criteria 1
  max_iterations 25
  concurrent_evaluations 4
  link_files topology_optimization_mech_ex6.exo
  copy_files target_data.csv analysis_driver.sh runmorphorm.py analysis.temp postprocess.
↪py
end study
```

The `concurrent_evaluations` keyword is used to set the number of concurrent response evaluations, i.e., how many simulation codes are launched concurrently to evaluate the design criteria for the outer-level optimization problem. The `link_files` keyword is used to link large files to the `run.{itr}` directories. The `run.{itr}` directories are created in the work directory (`workdir`). The work directory is created in the directory from where the optimization study is launched. The `{itr}` keyword is replaced with the numerical value for the major outer-level optimization iteration responsible for generating the results.

Inner Optimizer

The following input deck setup is for the inner-level topology optimization problem:

Listing 3.2.8: Input deck setup for the inner-level topology optimization problem.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
```

(continues on next page)

(continued from previous page)

```
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type mechanical_compliance
end criterion

begin criterion 2
  type inverse
  criterion_ids 5
  criterion_weights 1
  target_data target_data.csv
  degree_of_freedom dispz
end criterion

begin criterion 5
  type state_misfit
  transform exponential
end criterion

begin criterion 9
  type volume
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1 2 3
  linear_solver 1
  blocks 1
  output 1
end scenario

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 2
  services 2
  scenarios 1
end objective

begin output 1
  service 2
  output_frequency 20
  native_service_output true
```

(continues on next page)

(continued from previous page)

```
  data dispx dispy dispz vonmises
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name symm_plane
  degree_of_freedom dispx
  value 0.
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
  location_name le
  degree_of_freedom dispz
  value 0.
end boundary_condition

begin load 1
  type concentrated_load
  location_type nodeset
  location_name actuator1
  direction z
  value {load1}
end load

begin load 2
  type concentrated_load
  location_type nodeset
  location_name actuator2
  direction z
  value {load2}
end load

begin constraint 1
  criterion 9
  relative_target 0.3
  type less_than
  service 2
  scenario 1
end constraint

begin constraint 2
```

(continues on next page)

(continued from previous page)

```
  criterion 1
  absolute_target 3e5
  type less_than
  service 2
  scenario 1
end constraint

begin block 1
  material psudoti
  name design_area
end block

begin material psudoti
  material_model isotropic_linear_elastic
  poissons_ratio .33
  youngs_modulus 325e9
end material

begin study
  method topology
  discretization density
  initialize_method uniform
  initial_density_value 1.0

  max_iterations 50
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 2.
  projection_type tanh
  filter_projection_start_iteration 100
  filter_projection_update_interval 5
  filter_use_additive_continuation true
  filter_heaviside_min 1
  filter_heaviside_max 100
  filter_heaviside_update 2

  normalize_in_aggregator false
  fixed_exodus_sideset_names lower_surf_ss
  fixed_exodus_nodeset_names actuator1 actuator2
end study

begin mesh
  name topology_optimization_mech_ex6.exo
end mesh
```

The input deck in Listing 3.2.8 is referred in this manuscript as the `template input deck`. Given a trial set of actuating forces, the template input deck is used to create the input deck for the inner-level optimization study. The `aprepro` application [Sjaardema] [Sjaardema and USDOE] is used to do the parameter substitutions for the `{load1}` and `{load2}` keywords, i.e., parameter descriptors, at runtime. The parameter descriptors **must** be written within brackets, i.e., `{}`. The `aprepro` tool interprets a parameter descriptor within brackets as a parameter that must be replaced with a numerical value at runtime.

Run Study

To run the multi-level topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_mech_ex6.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_mech_ex6.i` is the *input deck* for the multi-level topology optimization study. The design study should run after pressing the `Enter` key.

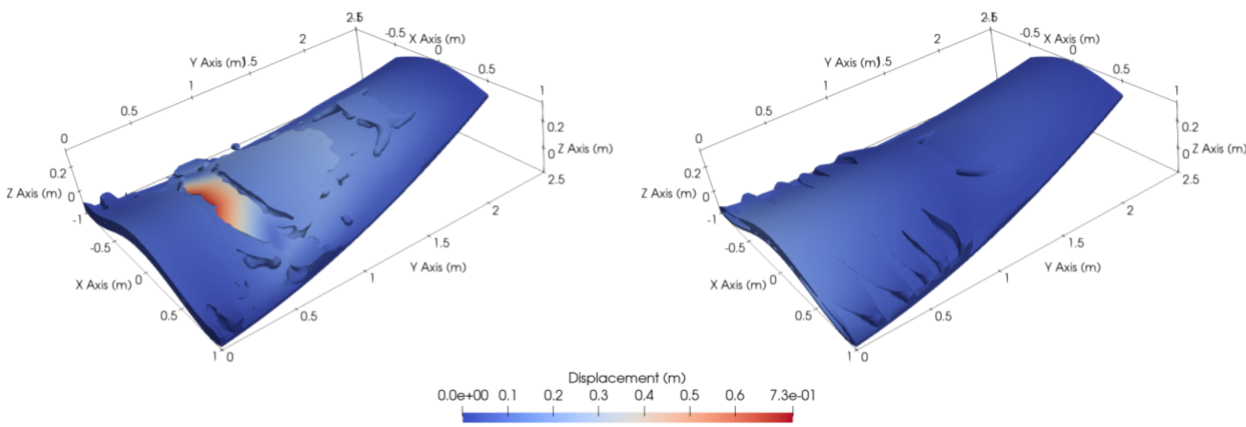

Fig. 3.2.13: The displacement field for the best (left pane) and the worst (right pane) design candidates. A total of 35 design candidates were found in the initial design exploration.

Results

A multi-level topology optimization study is explored to find suitable design candidates to match the desired deformation profile. The outer-level optimizer searches for the actuating forces while the inner-level optimizer optimizes the material layout. The goal is to find the combination of actuating forces and material layout that matches the desired deformation profile. Fig. 3.2.13 shows the displacement field for the best (left pane) and the worst (right pane) design candidates. These design candidates were found in the initial design exploration. The corresponding actuating forces for the two design candidates are shown in Table 3.2.3. The mass reduction goal of seventy percent was achieved for all 35 design candidates.

Table 3.2.3: Actuating forces for the best and worst design candidates. The shape matching error (Eq.3.2.21) is also tabulated.

Design Candidate	Load 1 (N)	Load 2 (N)	Shape Matching Error
Best	17377.0	7921.0	1.29
Worst	45136.0	43531.0	2.07

The initial design exploration produced 35 design candidates. The response surface is shown in Fig. 3.2.14. The location of the best design candidate on the response surface is marked with a pink circle. A subsequent multi-level optimization study could be done around this location to refine the actuating forces and material layout. However, this study is not done in this example.

A subsequent topology optimization study is done using the actuating forces that produced the best deformation profile match to refine the material layout. Thus, the actuating forces are assumed known in the second topology optimization study and only the material layout is optimized. Fig. 3.2.15 shows the optimized material layout results from the second

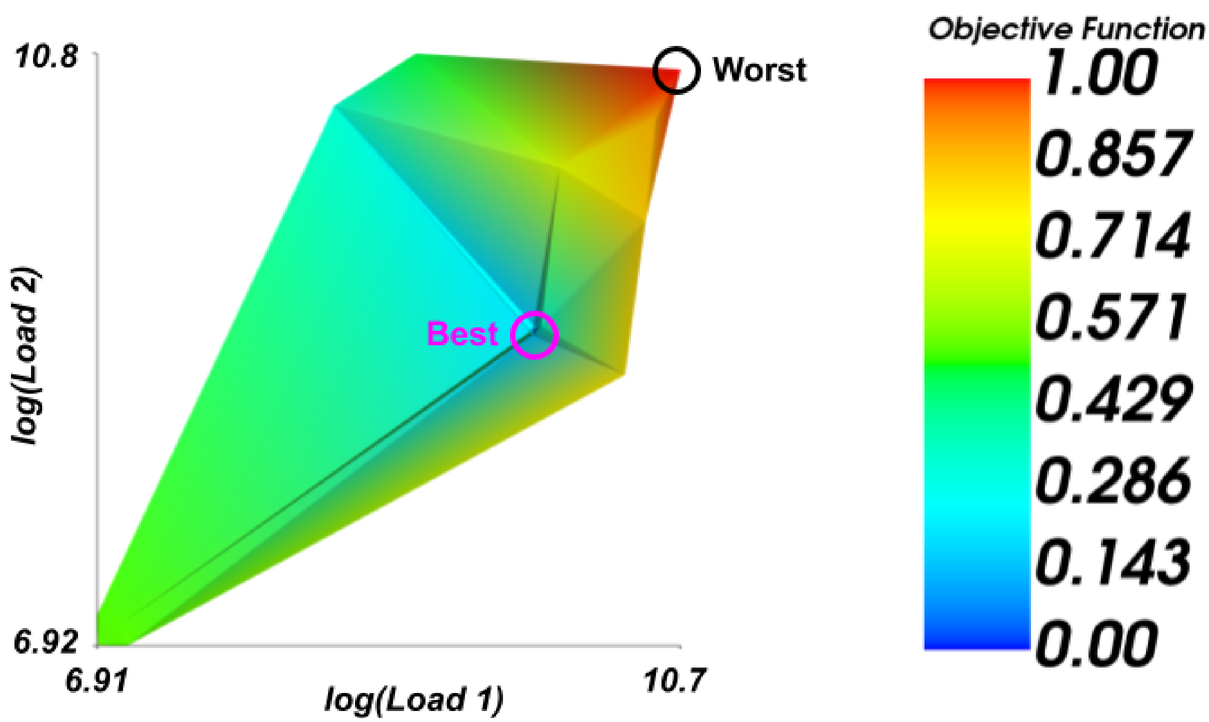

Fig. 3.2.14: The normalized response surface computed with the 35 design candidates. The location of the best design candidate on the response surface is marked with a pink circle. The location of the worst design candidate on the response surface is marked with a black circle.

topology optimization study. The displacement field computed with the optimized material layout is also plotted. The error of the inverse objective function (Eq.3.2.21) was 1.02, which is a twenty percent reduction in comparison to the error obtained in the initial multi-level design study. Finally, the mass reduction goal of seventy percent was achieved.

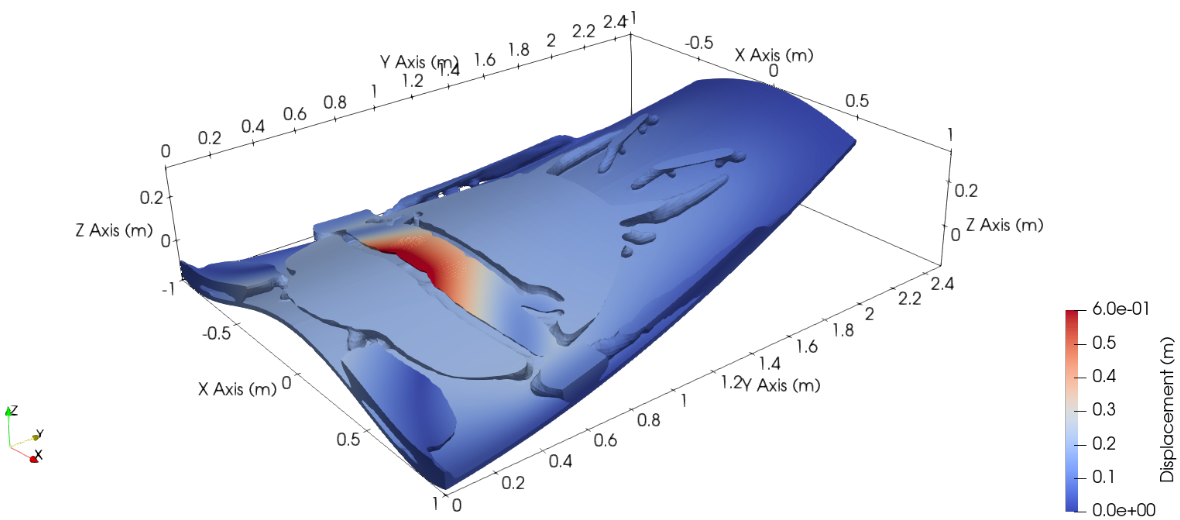

Fig. 3.2.15: The displacement field plotted on the optimized material layout found in the second topology optimization study.

3.3 Thermal

The following topology optimization examples describe how to use density-based topology optimization methods to optimize the layout of a fixed amount of material such that thermal design criteria are improved. The examples create a suite of tutorials which summarize the main concepts for solving thermal topology optimization problems.

3.3.1 Example 1: Compliance Minimization

This thermal topology optimization example seeks to find a lightweight bracket such that the heat transfer performance in the bracket is improved and a mass budget is satisfied.

Model Definition

Fig. 3.3.1 shows the computational domain, finite element mesh, and boundary conditions used for the thermal topology optimization study. The Dirichlet and Neumann boundary conditions and the non-optimizable regions for the study are:

1. Dirichlet boundary conditions
   a. The purple surfaces are kept at 310.9278 K, see the right pane in Fig. 3.3.1.
2. Neumann boundary conditions
   a. A thermal flux ($\mathrm{W/m^2}$) with magnitude -100 is applied at the cyan surface, see the right pane in Fig. 3.3.1.
3. Non-optimizable regions
   a. The red regions are defined as non-optimizable regions, see the right pane in Fig. 3.3.1.

The material thermal conductivity constant is set to 235.0 $\mathrm{W/(m \cdot K)}$.

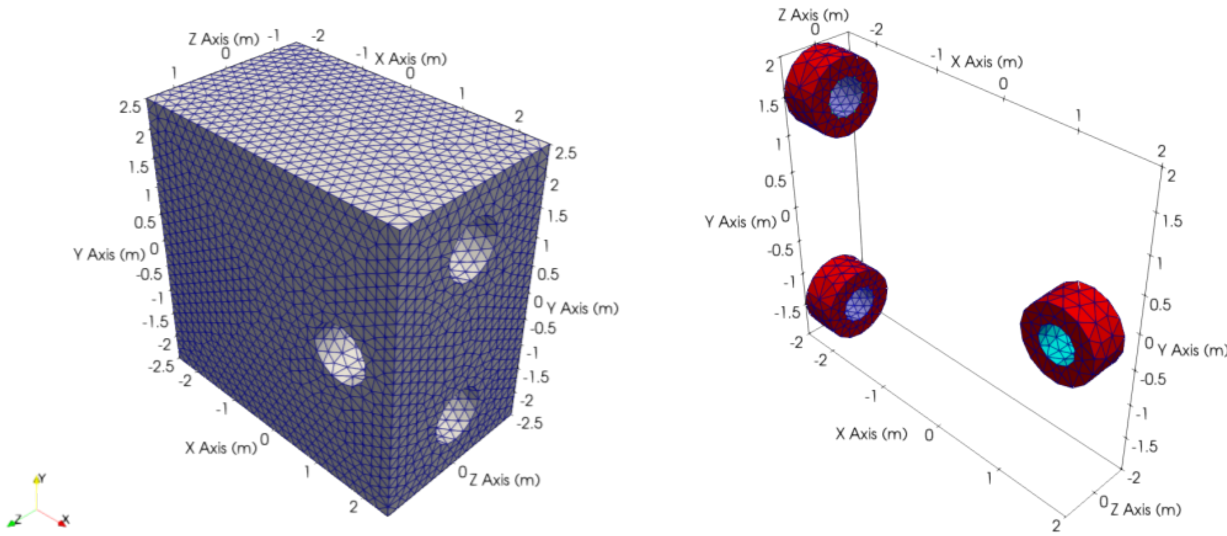

Fig. 3.3.1: The computational domain and the finite element mesh for the thermal topology optimization study are shown in the left pane. The boundary conditions used for the topology optimization study are shown in the right pane. The temperature (kelvin) at the purple surfaces is kept at 310.9278 K. A thermal flux of $-100\ \mathrm{W/m^2}$ is applied at the cyan surface. The red regions are set to non-optimizable regions, i.e., the optimizer cannot add nor removed material at these locations.

Formulation

The thermal topology optimization problem seeks to improve the heat transfer performance in the bracket subject to a mass budget. Mathematically, the thermal topology optimization problem of interest is defined as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_z}} \quad & \alpha f(\theta(\mathbf{z}),\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{R}(\theta(\mathbf{z}),\mathbf{z}) = 0 \\ & h(\mathbf{z}) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\dots,N_z \end{aligned} \tag{3.3.1}$$

where $\theta(\mathbf{z})$ is the temperature, which depends on the vector of design variables $\mathbf{z}$ of size N_z. $f(\theta(\mathbf{z}),\mathbf{z})$ is the thermal compliance criterion, which is defined as:

$$f(\theta(\mathbf{z}),\mathbf{z}) = \int_\Omega \kappa(\mathbf{z})\ \theta^2\ d\Omega \tag{3.3.2}$$

where Ω is the design domain. The material thermal conductivity $(\mathrm{W/m\cdot K})$ is defined as a positive material constant, i.e., $\kappa > 0.0$. The thermal compliance criterion is weighted by the scalar $\alpha = 1.0$. $\mathbf{R}(\theta(\mathbf{z}),\mathbf{z})$ is the residual vector for the linear thermal physics. The volume budget constraint $h(\mathbf{z})$ is defined as:

$$h(\mathbf{z}) = \int_\Omega \mathbf{z}\ d\Omega. \tag{3.3.3}$$

The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum feature size is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.3.1.

Input Deck

The input deck setup for the thermal topology optimization study is:

Listing 3.3.1: Input deck setup for the thermal topology optimization study.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type thermal_compliance
  minimum_ersatz_material_value 1e-9
end criterion

begin criterion 2
  type volume
end criterion

begin scenario 1
  physics steady_state_thermal
  service 2
  dimensions 3
  loads 1
  boundary_conditions 1 2
  material 1
  minimum_ersatz_material_value 1e-9
end scenario

begin objective
  type weighted_sum
  criteria 1
  services 2
  scenarios 1
  weights 1
end objective

begin boundary_condition 1
  type fixed_value
```

(continues on next page)

(continued from previous page)

```
  location_type nodeset
  location_name ns_1
  degree_of_freedom temp
  value 310.9278
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom temp
  value 310.9278
end boundary_condition

begin load 1
  type thermal_flux
  location_type sideset
  location_name ss_1
  value -1e2
end load

begin constraint 1
  criterion 2
  relative_target .3
  type less_than
  service 1
end constraint

begin block 1
  material 1
end block

begin block 2
  material 1
end block

begin material 1
  material_model isotropic_linear_thermal
  thermal_conductivity 235
end material

begin output
  service 2
  output_frequency 5
  data temperature thermal_flux
  native_service_output true
end output

begin study
  method topology
  discretization density
  initial_density_value 1.0
```

(continues on next page)

(continued from previous page)

```
  max_iterations 100
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 2
  projection_type tanh
  filter_projection_start_iteration 100
  filter_projection_update_interval 5
  filter_use_additive_continuation true
  filter_heaviside_min 1
  filter_heaviside_max 100
  filter_heaviside_update 2

  fixed_block_ids 2
end study

begin mesh
  name topology_optimization_thermal_ex1.exo
end mesh
```

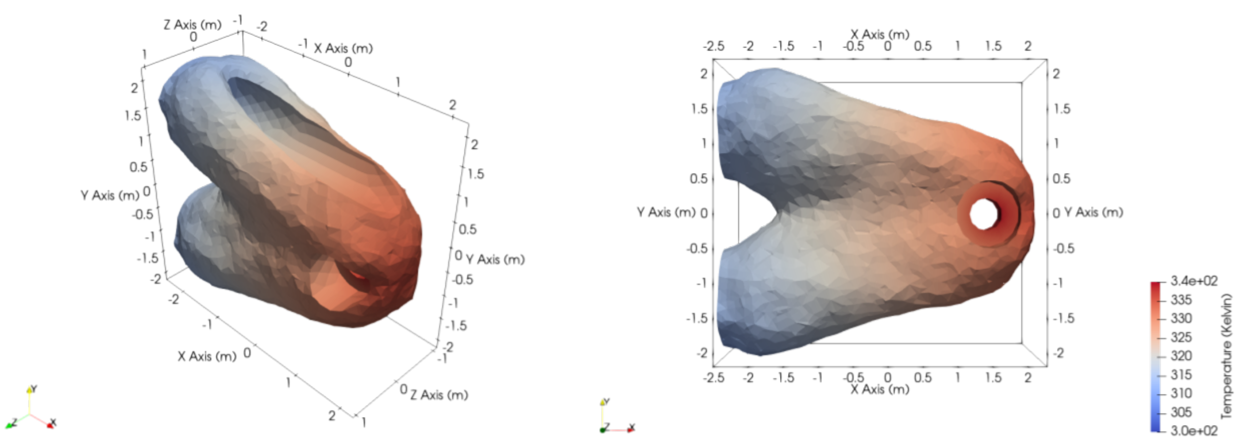

Fig. 3.3.2: Temperature field for the topology optimized bracket.

Run Study

To run the thermal topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_thermal_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_thermal_ex1.i` is the *input deck* for the thermal topology optimization study. The design study should run after pressing the `Enter` key.

Results

Fig. 3.3.2 shows the temperature field for the topology optimized bracket. The 70 percent mass reduction goal was achieved while improving the heat transfer performance of the bracket.

3.3.2 Example 2: Local Thermal Flux Constraints

This thermal topology optimization study seeks to find a lightweight structure such that a thermal flux limit is met at each material point. This class of topology optimization problems becomes computationally intractable if the thermal flux limit at each material point are not properly modeled. Indeed, at a minimum, the simulation code must be called $1+N_g$ times at each major optimization iteration k, where N_g denotes the number of thermal flux constraints. Each call to the simulation code requires the solution of a large linear system of equations. The augmented Lagrangian method is employed in this example to efficiently solve the topology optimization problem with a large number of local thermal flux constraints.

Model Definition

Fig. 3.3.1 shows the computational domain, finite element mesh, and boundary conditions used for the topology optimization study. The Dirichlet and Neumann boundary conditions and the non-optimizable regions for the topology optimization study are:

1. Dirichlet boundary conditions
   a. The purple surfaces are kept at 310.9278 K, see the right pane in Fig. 3.3.1.
2. Neumann boundary conditions
   a. A thermal flux ($\mathrm{W/m^2}$) with magnitude -100 is applied at the cyan surface, see the right pane in Fig. 3.3.1.
3. Non-optimizable regions
   a. The red regions are defined as non-optimizable regions, see the right pane in Fig. 3.3.1.

The material thermal conductivity constant is set to $235.0\ \mathrm{W/(m \cdot K)}$.

Formulation

The topology optimization problem seeks to find a lightweight structure such that a thermal flux limit is met at each material point. The topology optimization problem under consideration can be formulated as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\theta(\mathbf{z}),\mathbf{z}\right) \\ \text{s.t.} \quad & \mathbf{R}\left(\theta(\mathbf{z}),\mathbf{z}\right) = 0 \\ & g_j\left(\theta(\mathbf{z}),\mathbf{z}\right) \leq 0,\ j = 1,\dots,N_g \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\dots,N_z \end{aligned} \tag{3.3.4}$$

where $N_f = 2$ is the number of objective functions and N_g is the number of local thermal flux constraints. $\theta(\mathbf{z})$ is the temperature, which depends on the vector of design variables $\mathbf{z}$ of size N_z.

The first objective function is defined by the volume criterion:

$$f_{l=1}(\mathbf{z}) = \int_\Omega \mathbf{z}\, d\Omega, \tag{3.3.5}$$

where Ω is the design domain. The second objective function is defined by the thermal compliance criterion:

$$f_{l=2}(\theta(\mathbf{z}),\mathbf{z}) = \int_\Omega \kappa(\mathbf{z})\ \theta^2\ d\Omega, \tag{3.3.6}$$

where $\kappa(\mathbf{z})$ is the material thermal conductivity $(\mathrm{W/m \cdot K})$, defined as a positive constant, i.e., $\kappa > 0.0$. Notice that $\kappa(\mathbf{z})$ depends on the material layout parameterization. To encourage optimal heat transfer performance, Eq.3.3.5 and Eq.3.3.6 are weighted by scalars $\alpha_{l=1} = 1.0$ and $\alpha_{l=2} = 0.01$, respectively. The residual vector for the elliptic thermal physics is denoted with $\mathbf{R}(\theta(\mathbf{z}), \mathbf{z})$. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum feature size is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.3.4.

The optimization problem in Eq.3.3.4 becomes computationally intractable if the nonlinear constraints g_j are not properly managed. An augmented Lagrangian method is used instead to efficiently model the large number of nonlinear constraints in Eq.3.3.4. At each major optimization iteration k, a sequence of optimization problems is solved to find an optimal and feasible material layout solution. Mathematically, the topology optimization problem with local thermal flux constraints in Eq.3.3.4 is recast as:

$$
\begin{aligned}
\min_{\mathbf{z} \in \mathbb{R}^{N_{\mathbf{z}}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\theta(\mathbf{z}), \mathbf{z}\right) + \frac{\beta}{N_g} \sum_{j=1}^{N_g} \left[\gamma_j^{(k)} \hat{g}_j\left(\theta(\mathbf{z}), \mathbf{z}\right) + \frac{\mu_j^{(k)}}{2} \hat{g}_j\left(\theta(\mathbf{z}), \mathbf{z}\right)^2 \right] \\
\text{s.t.} \quad & \mathbf{R}\left(\theta(\mathbf{z}), \mathbf{z}\right) = 0 \\
& \underline{z}_i \leq z_i \leq \bar{z}_i, \; i = 1, \ldots, N_z,
\end{aligned}
\tag{3.3.7}
$$

where $\beta > 0$ and

$$
\hat{g}_j\left(\theta(\mathbf{z}), \mathbf{z}\right) = \left(g_j\left(\theta(\mathbf{z}), \mathbf{z}\right), -\frac{\gamma_j^{(k)}}{\mu_j^{(k)}} \right), \tag{3.3.8}
$$

$$
\gamma_j^{(k+1)} = \gamma_j^{(k)} + \mu_j^{(k)} \hat{g}_j\left(\theta(\mathbf{z}), \mathbf{z}\right), \tag{3.3.9}
$$

$$
\mu_j^{(k+1)} = \left(\alpha \mu_j^{(k)}, \mu_{max} \right), \; \alpha > 1. \tag{3.3.10}
$$

The augmented Lagrangian problem in Eq.3.3.7 is solved until a convergence criteria is met. The parameters $\gamma_j^{(k)}$ and $\mu_j^{(k)}$ are the j-th Lagrange multiplier and the j-th penalty, respectively. These two parameters are associated with the j-th nonlinear constraints $g_j^{(k)}$.

Input Deck

The input deck setup for the thermal topology optimization study with local thermal flux constraints is:

Listing 3.3.2: Input deck setup for the thermal topology optimization study with local thermal flux constraints.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 10
end service
```

(continues on next page)

(continued from previous page)

```
begin criterion 1
  type local_constraint
  limits 5e3
  local_measures thermal_flux
  initial_penalty 0.1
  minimum_ersatz_material_value 1e-6
end criterion

begin criterion 2
  type volume
end criterion

begin criterion 3
  type thermal_compliance
  minimum_ersatz_material_value 1e-6
end criterion

begin scenario 1
  physics steady_state_thermal
  service 2
  dimensions 3
  loads 1
  boundary_conditions 1 2
  material 1
  minimum_ersatz_material_value 1e-6
end scenario

begin objective
  type weighted_sum
  criteria 1 2 3
  services 2 2 2
  scenarios 1 1 1
  weights 1e2 1 1e-2
end objective

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name ns_1
  degree_of_freedom temp
  value 310.9278
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom temp
  value 310.9278
end boundary_condition
```

(continues on next page)

(continued from previous page)

```
begin load 1
  type thermal_flux
  location_type sideset
  location_name ss_1
  value -5e3
end load

begin block 1
  material 1
  location_name block_1
end block

begin block 2
  material 1
  location_name block_2
end block

begin material 1
  material_model isotropic_linear_thermal
  thermal_conductivity 235
end material

begin output
  service 2
  output_frequency 10
  data temperature thermal_flux
  native_service_output true
end output

begin study
  method topology
  discretization density
  initial_density_value 1.0

  max_iterations 200
  optimization_algorithm mma
  mma_move_limit 0.1
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 2.0
  fixed_block_ids 2
end study

begin mesh
  name topology_optimization_thermal_ex2.exo
end mesh
```

Run Study

To run the thermal topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_thermal_ex2.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_thermal_ex2.i` is the *input deck* for the thermal topology optimization study with local thermal flux constraints. The design study should run after pressing the `Enter` key.

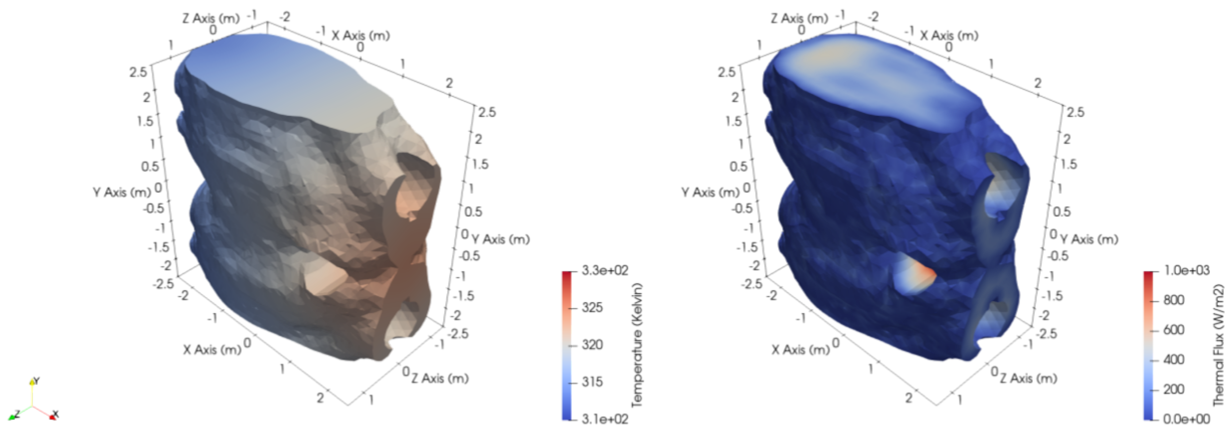

Fig. 3.3.3: Temperature (left pane) and thermal flux (right pane) fields for the topology optimized bracket. The topology optimizer produced an optimized bracket that met the thermal flux limit at most material points while removing excess material.

Results

Fig. 3.3.3 shows the temperature and thermal flux fields for the topology optimized bracket. The thermal flux limit is met at most material points, with the exception of the material points located near the surface where the thermal flux is applied, which is a non-optimizable region. Furthermore, a 32 percent mass reduction was achieved while improving the heat transfer performance of the bracket.

3.4 Thermomechanics

The following topology optimization examples demonstrate how to use density-based topology optimization to optimize the distribution of a fixed amount of material such that thermal and mechanical design criteria are improved. The examples create a suite of tutorials which summarize the main concepts for solving thermomechanical topology optimization problems.

3.4.1 Example 1: Compliance Minimization

This thermomechanical topology optimization example seeks to maximize the stiffness of the structure and to improve the heat transfer performance of the structure. Modifying the mass budget leads to a different material distribution. Therefore, the topology optimizer must balance several conflicting design criteria: stiffness, heat transfer, and mass.

Model Definition

Fig. 3.4.1 shows the computational domain of the original aircraft. The design domain is symmetric about the plane at $X = 0$ and is made of a linear elastic material. Therefore, only half of the domain is modeled for the thermomechanical topology optimization study. The design is optimized for thermal and mechanical environments. The boundary conditions for the topology optimization study are:

1. Dirichlet boundary conditions

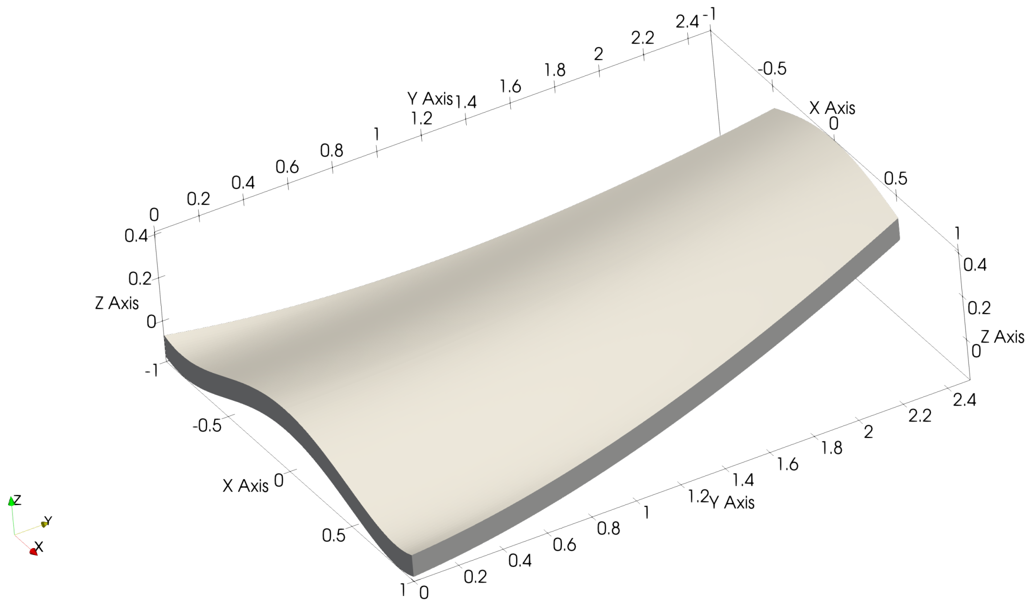

Fig. 3.4.1: The computational domain for the topology optimization study.

a. The X, Y, and Z displacements (m) on the leading edge, i.e., the surface defined by the plane at $Y = 2.5$, are fixed.

b. The X displacements on the symmetry surface, i.e., the surface defined by the plane at $X = 0$, are fixed.

c. The Y displacements along the edge with starting and ending points $(1, 0, -0.1667)$ and $(0.5, 2.5, 0.2083)$ are fixed.

d. The top surface is kept at $700°$C.

e. The bottom surface is kept at $1200°$C.

f. The front surface defined by the plane at $Y = 2.5$ is kept at $3000°$C.

2. External forces

   a. A concentrated force (N) with components $(0.0, 0.0, -6200)$ is applied at the top red surface.

   b. A concentrated force with components $(0.0, 0.0, -3475)$ is applied at the top blue surface.

3. Non-optimizable regions

   a. The nodes on the bottom surface are set to non-optimizable nodes.

   b. The nodes on the top red and blue surfaces are set to non-optimizable nodes.

The elastic modulus is set to 120 GPa and Poisson's ratio is set to 0.34. The material conductivity constant is set to 17 W/m^2 $\cdot^\circ$ C and the coefficient of thermal expansion is set to 1.0×10^{-8} m/m $\cdot^\circ$ C.

Formulation

Given a mass budget, the optimization problem seeks to find a lightweight and stiff structure while improving the heat transfer performance of the structure. These design goals are achieved by minimizing the mechancial and thermal compliance criteria subject to a volume budget constraint. Mathematically, the thermomechanical topology optimization

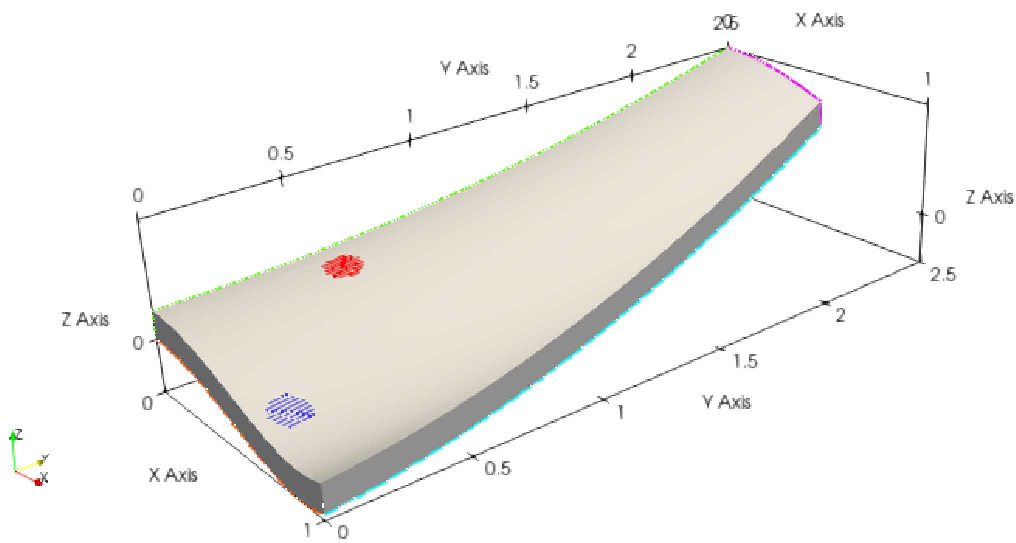

Fig. 3.4.2: The computational domain used for the topology optimization study. The red and blue circles on the top surface are the locations where the concentrated forces are applied. The Y displacements (m) on the cyan edge are fixed. The the X, Y, and Z displacements on the front pink surface, i.e., surface on the plane at $Y = 2.5$, are fixed. The X displacements on the green surface, i.e., the symmetry plane, are fixed. The orange surface, i.e., bottom surface, is kept at $1200°\mathrm{C}$. The surface defined by the plane at $Y = 2.5$ is kept at $3000°\mathrm{C}$. The top surface is kept at $700°\mathrm{C}$. A concentrated force (N) with components $(0.0, 0.0, -6200)$ is applied at the top red surface. A concentrated force with components $(0.0, 0.0, -3475)$ is applied at the top blue surface.

problem is defined as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_z}} \quad & \alpha_{\mathbf{u}} f_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z}) + \alpha_\theta f_\theta(\theta(\mathbf{z}),\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{u}(\mathbf{z}),\theta(\mathbf{z}),\mathbf{z}) = 0 \\ & h(\mathbf{z}) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\ldots,N_z \end{aligned} \tag{3.4.1}$$

where $\mathbf{u}(\mathbf{z})$ and $\theta(\mathbf{z})$ are the displacement and temperature fields, respectively. The displacements and temperature fields depend on the vector of design variables (densities) $\mathbf{z}$ of size N_z. $f_{\mathbf{u}}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the mechanical compliance criterion, which is weighted by the scalar $\alpha_{\mathbf{u}} \geq 0$. $f_\theta(\theta(\mathbf{z}),\mathbf{z})$ is the thermal compliance criterion, which is weighted by the scalar $\alpha_\theta \geq 0$. $\mathbf{R}(\mathbf{u}(\mathbf{z}),\theta(\mathbf{z}),\mathbf{z})$ is the residual equation for the elliptic thermomechanical physics and $h(\mathbf{z})$ is the volume budget constraint. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively. A minimum length scale is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.4.1.

Input Deck

The following text snippet is the input deck setup for the thermomechanical topology optimization study:

Listing 3.4.1: Input deck setup for the thermomechanical topology optimization study.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type thermomechanical_compliance
  thermal_weighting_factor 1.0
  mechanical_weighting_factor 1.0
end criterion

begin criterion 2
  type volume
end criterion

begin scenario 1
  service 2
  dimensions 3
  physics steady_state_thermomechanics
  loads 1 2
  boundary_conditions 1 2 3 4 5 6
  linear_solver 1
  newton_raphson 1
  blocks 1
```

(continues on next page)

(continued from previous page)

```
  output 1
end scenario

begin newton_raphson 1
  residual thermomechanical
  residual_tolerance 1e-8
  increment_tolerance 1e-8
  max_iterations 100
  num_increments 1
end newton_raphson

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 1
  services 2
  scenarios 1
end objective

begin constraint 1
  criterion 2
  relative_target 0.3
  type less_than
  service 1
  scenario 1
end constraint

begin output 1
  service 2
  output_data true
  data dispx dispy dispz temperature stress vonmises thermal_flux
  native_service_output true
  output_frequency 25
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name symm_plane
  degree_of_freedom dispx
  value 0.
```

(continues on next page)

(continued from previous page)

```
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
  location_name le
  degree_of_freedom dispz
  value 0.
end boundary_condition

begin boundary_condition 4
  type fixed_value
  location_type nodeset
  location_name lower_surf
  degree_of_freedom temp
  value 1200.
end boundary_condition

begin boundary_condition 5
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom temp
  value 3000.
end boundary_condition

begin boundary_condition 6
  type fixed_value
  location_type nodeset
  location_name top_surf
  degree_of_freedom temp
  value 700.
end boundary_condition

begin load 1
  type concentrated_load
  location_type nodeset
  location_name actuator1
  direction z
  value -6200
end load

begin load 2
  type concentrated_load
  location_type nodeset
  location_name actuator2
  direction z
  value -3475
end load

begin block 1
  material 1
```

(continues on next page)

(continued from previous page)

```
  name design_area
end block

begin material 1
  material_model isotropic_linear_thermoelastic
  poissons_ratio 0.34
  youngs_modulus 120e9
  thermal_expansivity 1e-8
  thermal_conductivity 17.
  reference_temperature 0.
end material

begin study 1
  method topology
  initialize_method uniform
  initial_density_value 1.0
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true
  max_iterations 100
  filter_radius_scale 2.
  fixed_exodus_sideset_names lower_surf_ss
  fixed_exodus_nodeset_names actuator1 actuator2
end study

begin mesh
  name topology_optimization_tmech_ex1.exo
end mesh
```

Run Study

To run the thermomechanical topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_tmech_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_tmech_ex1.i` is the *input deck* for the thermomechanical topology optimization study. The design study should run after pressing the `Enter` key.

Restart Study

The topology optimization results from a previous topology optimization study can be to set the initial guess for a new topology optimization study. The `study block` in the new input deck, i.e., input deck for the new topology optimization study, must be modified as follows:

```
begin study 1
  method topology
  initialize_method uniform
  initial_density_value 1.0
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true
  max_iterations 100
```

(continues on next page)

(continued from previous page)

```
  filter_radius_scale 2.
  fixed_exodus_sideset_names lower_surf_ss
  fixed_exodus_nodeset_names actuator1 actuator2

  restart_iteration 100
  initial_guess_field_name topology
  initial_guess_file_name engine_output.exo
end study
```

The remaining input deck blocks and their parameters stay the same as those used in Listing 3.4.1, unless some of the parameters need to be updated for the new topology optimization study, e.g., reduce the number of optimization iterations. The user is advised to consult the `User Manual` to learn more about how to use the input parameters for the restart application.

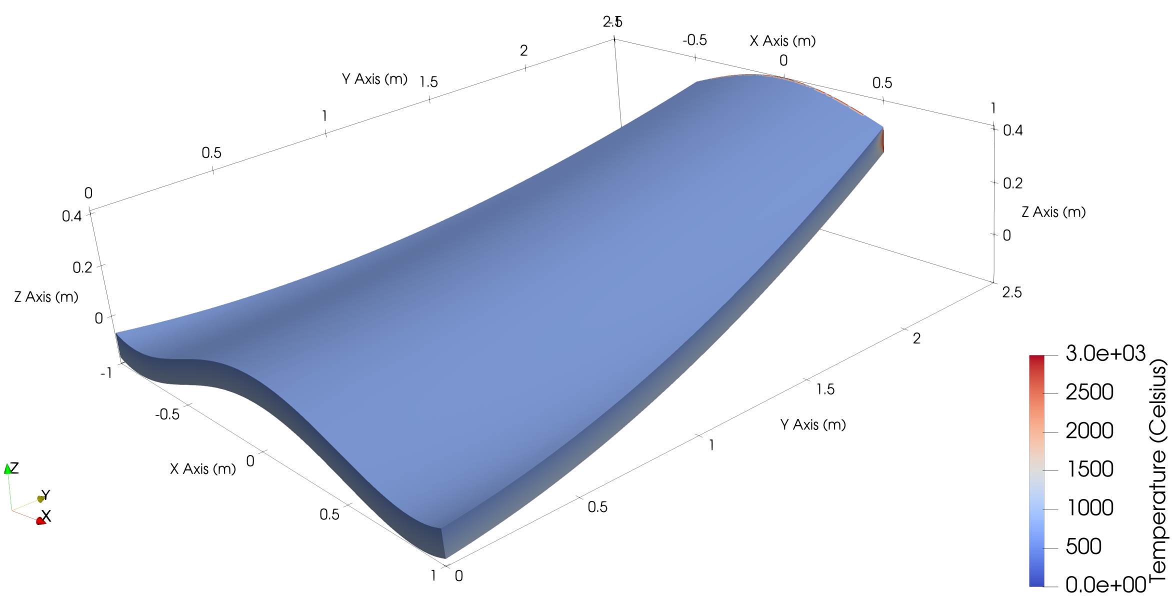

Fig. 3.4.3: Temperature $(^{\circ}\mathrm{C})$ distribution for the baseline design.

Results

Fig. 3.4.3 and Fig. 3.4.4 show the temperature and displacement fields for the baseline design. The topology optimization problem attempts to maximize the stiffness of the structure, improve the heat transfer performance, and meet a volume/mass budget.

The displacement and temperature fields for the topology optimized designs are shown in Fig. 3.4.5 and Fig. 3.4.6, respectively. The top row in Fig. 3.4.5 shows the material layout results and the corresponding displcement field obtained for the design study that equally prioritized the mechanical and thermal compliance criteria, i.e., $\alpha_{\mathbf{u}} = \alpha_\theta = 1$. The solution achieved the desired 70 percent mass reduction target while reducing the peak displacement from 0.12 m to 0.1088 m.

The middle row in Fig. 3.4.5 shows the material layout results and the displacement field for the design study that only prioritize the mechanical compliance criterion, i.e., $\alpha_{\mathbf{u}} = 1.0$ and $\alpha_\theta = 0.0$. The solution reached the 70 percent mass

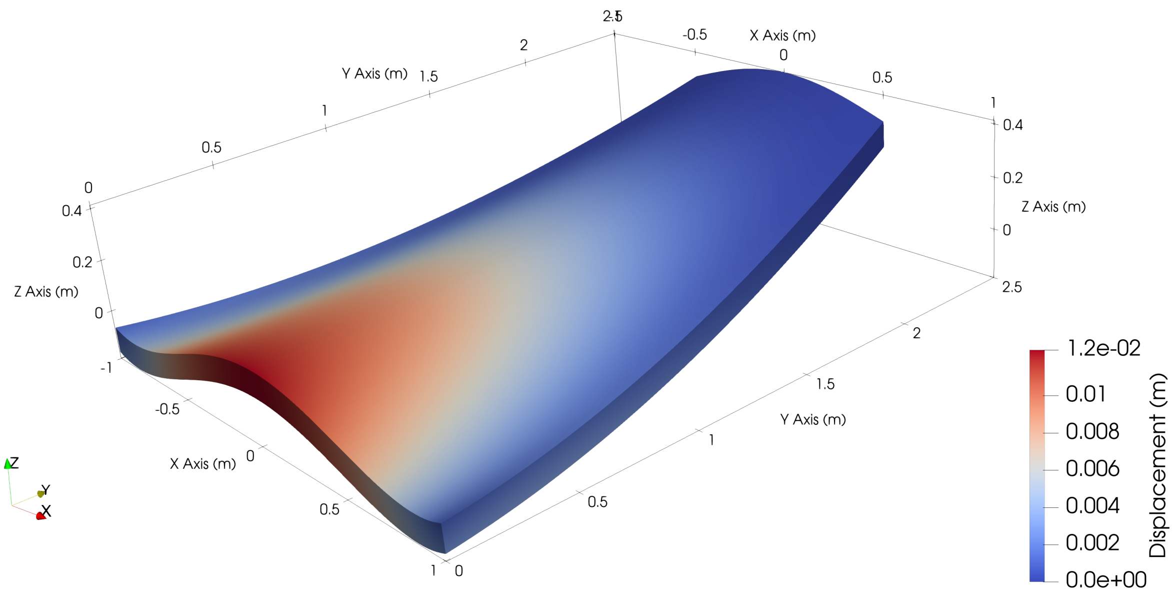

Fig. 3.4.4: Displacement (m) distribution for the baseline design.

reduction target. The peak displacement was reduced from 0.12 m to 0.03 m. Thus, the stiffer design was obtained when only the mechanical compliance criterion was considered for optimization.

The bottom row in Fig. 3.4.5 shows the material layout results for the design study that prioritized the thermal compliance criterion more than the mechanical compliance criterion, i.e., $\alpha_\theta = 1.$ and $\alpha_{\mathbf{u}} = 0.5$. This study produced the design with the highest peak displacement, i.e., 0.14 m, since the mechanical compliance criterion was deprioritized for this study. However, the design solution did reach the 70 percent mass reduction target. Finally, the topology optimization studies that gave a higher priority to the thermal compliance criterion produced designs with more sophisticated geometric features to improve heat transfer performance.

The topology optimization studies presented in this section highlight the impact that the objective function weights can have on the material layout. The next *thermomechanical topology optimization example* will investigate this observation.

3.4.2 Example 2: Multi-Dimensional Parameter Study

This thermomechanical topology optimization study seeks to maximize the stiffness of the structure, improve heat transfer performance, and meet a desired mass budget. *Example 1* explored how criterion prioritization changes impact the material layout results. *Example 2* will show users how to leverage the multi-dimensional parameter study application to find the best combination of criteria weights for a design study. The model definition and topology optimization formulation used in Example 2 is the same as those used in *Example 1*.

Input Deck

Example 2 requires two input decks. The first input deck is for the multi-dimensional parameter study. The second input deck is for the thermomechanical topology optimization study.

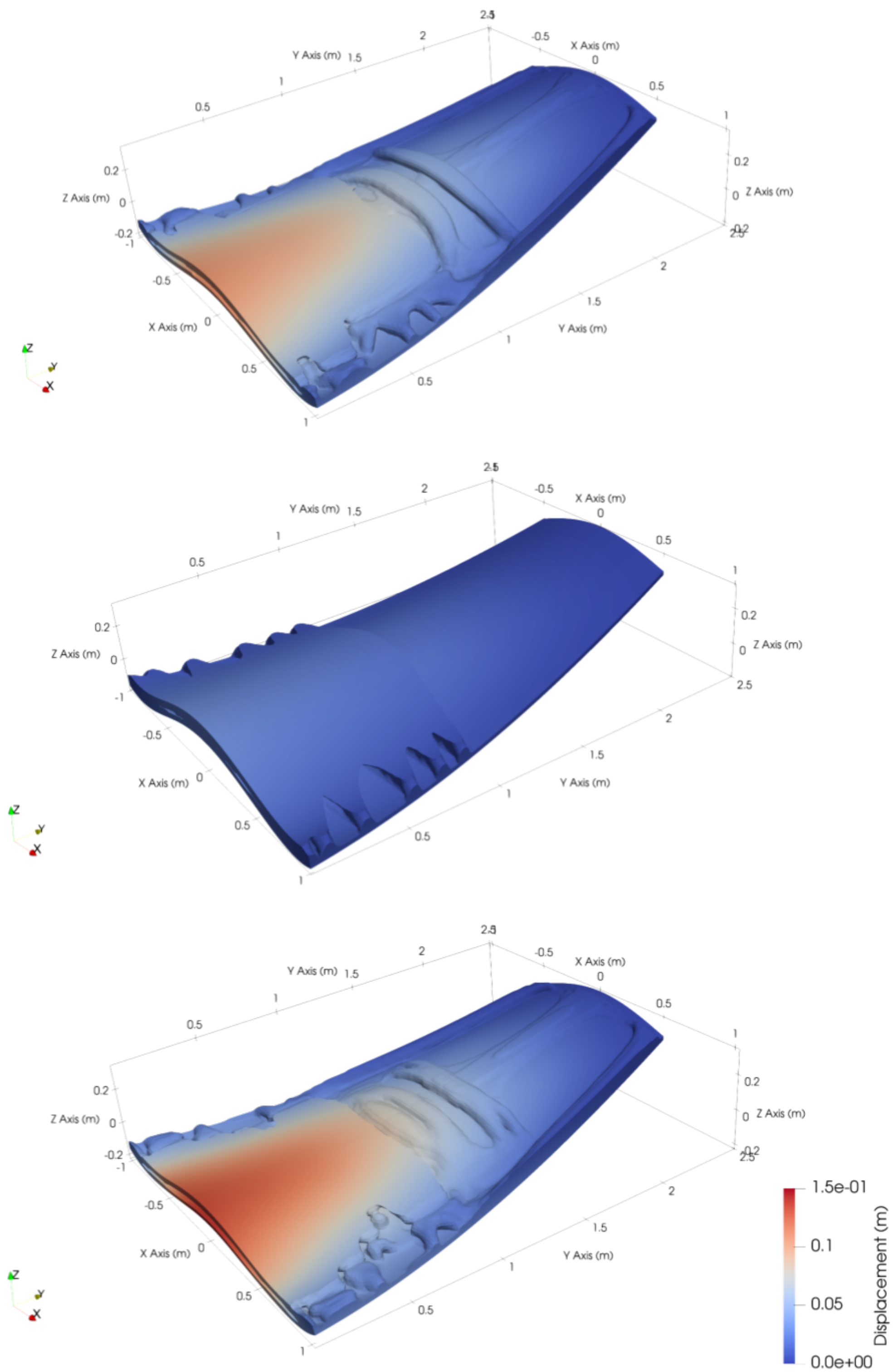

Fig. 3.4.5: Displacement (m) fields for the topology optimized structures. The top row shows the displacement field obtained for the design study that prioritized the mechanical and thermal compliance criteria equally, i.e., $\alpha_{\mathbf{u}} = \alpha_{\theta} = 1.0$. The middle row shows the displacement field for the design study that only prioritized the mechanical compliance criterion, i.e., $\alpha_{\mathbf{u}} = 1.0$ and $\alpha_{\theta} = 0.0$. The bottom row shows the displacement field for the design study that put higher priority on the thermal compliance criterion ($\alpha_{\theta} = 1.0$) rather than on the mechanical compliance criterion ($\alpha_{\mathbf{u}} = 0.5$).

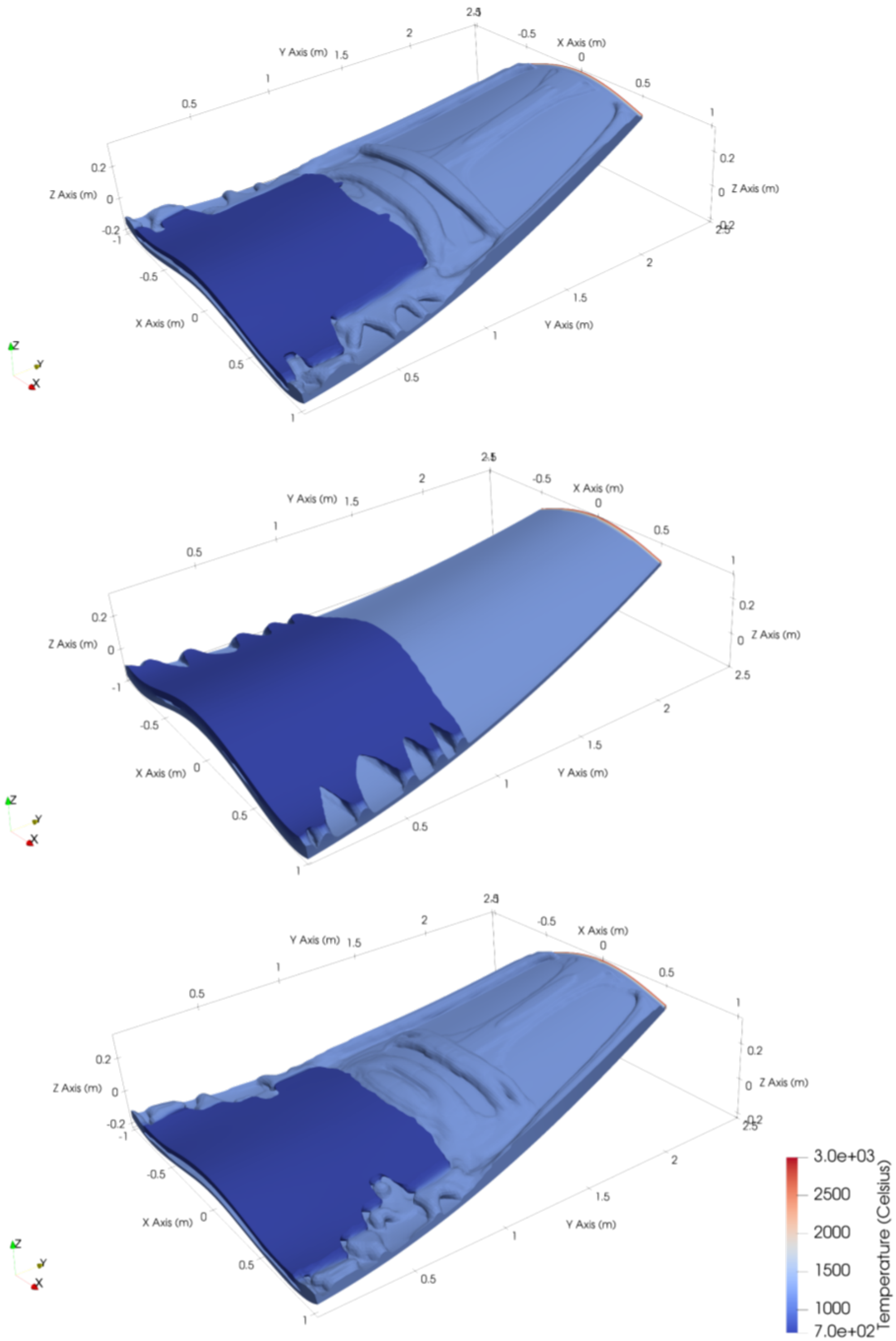

Fig. 3.4.6: Temperature (°C) fields for the topology optimized structures. The top row shows the temperature field for the design obtained with the study that equally prioritized the mechanical and thermal compliance criteria, i.e., $\alpha_{\mathbf{u}} = \alpha_{\theta} = 1.0$. The middle row shows the temperature field for the design obtained with the study that only prioritized the mechanical compliance criterion, i.e., $\alpha_{\mathbf{u}} = 1.0$ and $\alpha_{\theta} = 0.0$. The bottom row shows the temperature field for the design obtained with the study that put higher priority on the thermal compliance criterion ($\alpha_{\theta} = 1.0$) than on the mechanical compliance criterion ($\alpha_{\mathbf{u}} = 0.5$).

Parameter study

The following text snippet is the input deck setup for the multi-dimensional parameter study:

```
begin variables 1
  numvars 2
  type continuous_design
  initial_points 0.00  0.00
  lower_bounds   0.50  0.50
  upper_bounds   1.00  1.00
  descriptors     wt    wm
end variables

begin study
  method multidim_parameter_study
  variables 1
  num_criteria 2
  partitions 5 5
  link_files analysis_driver.sh runmorphorm.py topology_optimization_tmech_ex2.temp␣
↪postprocess.py
  concurrent_evaluations 4
end study
```

Please consult the `User Manual` to learn how to best set the input parameters for the multi-dimensional parameter study tool.

Topology optimization

The input deck setup for the thermomechanical topology optimization study was introduced in *Example 1*. However, the input deck for the must be slightly modified to enable parameter substitutions at runtime. For instance, the `criteria block 1` must be defined as:

```
begin criterion 1
  type thermomechanical_compliance
  thermal_weighting_factor {wt}
  mechanical_weighting_factor {wm}
end criterion
```

The scalar values for the `thermal_weighting_factor` and the `mechanical_weighting_factor` keywords have been replaced with the descriptors `{wt}` and `{wm}`, respectively. The other input blocks and their respective parameters are kept the same as those used for the *input deck used in Example 1*.

The final step is to create a template input deck. The template input deck is used to generate the input deck for the thermomechanical topology optimization study at runtime. The `aprepro` application [Sjaardema], [Sjaardema and USDOE] is used to perform the parameter substitutions for the `{wt}` and `{wm}` descriptors at runtime. The parameter descriptors **must** be written within brackets, i.e., `{}`. The `aprepro` tool interprets a parameter descriptor within brackets as a parameter that must be replaced with a numerical value at runtime.

Additional Scripts

Two additional scripts are needed to run a multi-dimensional parameter study. The first script is the `analysis driver script`, which performs the parameter substitutions, runs the topology optimization study, and postprocess the topology optimization results. The second script is the `postprocessing script`, which writes the file containing the objective and constraint function values from the topology optimization study.

Analysis driver

The analysis_driver.sh script contains the sequence of instructions for:

- performing the parameter substitutions,
- running the thermomechanical topology optimization study, and
- writing the file containing the objective and constraint function values.

The following text snippet shows the setup used in *Example 2* for the analysis_driver.sh script:

```
#!/bin/bash
aprepro $1 topology_optimization_tmech_ex2.temp topology_optimization_tmech_ex2.i
python runmorphorm.py topology_optimization_tmech_ex2.i
python postprocess.py
```

The first line instructs the operating system to use bash as the command interpreter. The second line is used to call the built-in aprepro application [Sjaardema], [Sjaardema and USDOE], which performs the parameter substitutions and creates the input deck for the topology optimization study. The third line is used to launch the thermomechanical topology optimization study. The last line is used to call the application that writes the file (results.out) containing the best objective and constraint function values computed by the material layout optimizer, i.e., topology optimization algorithm.

Postprocessing

The postprocess.py script is used to write the file (results.out) containing the objective and constraint function values, i.e., design criteria being investigated in the parameter study. The following is a typical setup for the postprocess.py script:

```
import itertools
# set filenames
#
tMyFileList=["opt_criteria_history.csv"]
# open output file
#
tOutFile = open("results.out", "w+")
# loop over files and read its data
#
for tMyFile in tMyFileList:
  # open file
  #
  with open(tMyFile) as tFile:
    # read lines
    #
    tLines = tFile.readlines()
    # loop over lines
    #
    for tLine in tLines:
      # split line
      #
      tMyList = tLine.split(', ')
      # write last criterion value to output file
      #
      tOutFile.write(tMyList[len(tMyList)-2])
      tOutFile.write("\n")
```

(continues on next page)

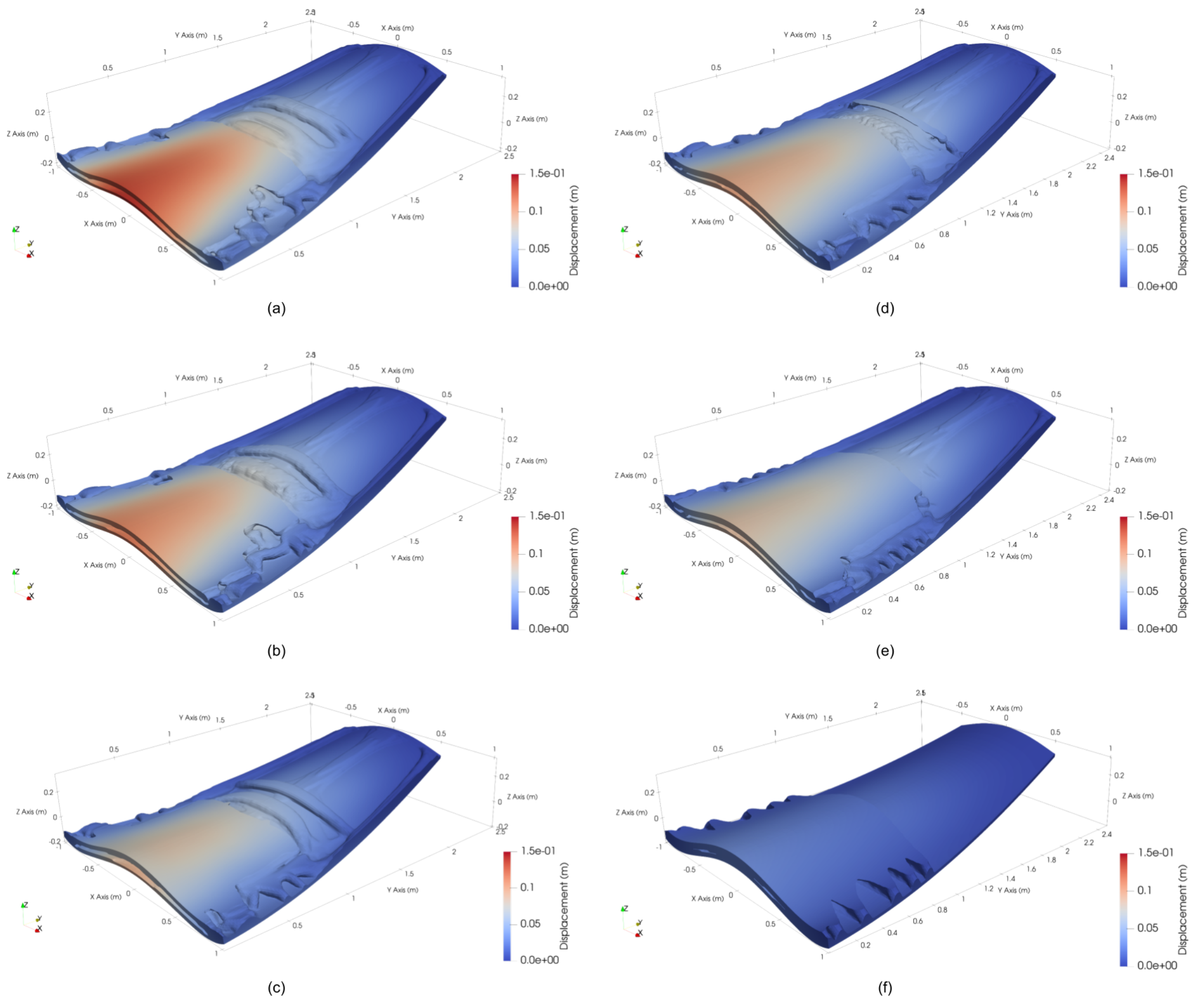

Fig. 3.4.7: Displacement (m) field for the topology optimized structures. The plot only shows six out of the twenty-five samples investigated with the multi-dimensional parameter study algorithm. The samples correspond to the following criteria weight pairs: (a) $\alpha_\theta = 1.0$ and $\alpha_{\mathbf{u}} = 0.5$, (b) $\alpha_\theta = 1.0$ and $\alpha_{\mathbf{u}} = 0.75$, (c) $\alpha_\theta = 1.0$ and $\alpha_{\mathbf{u}} = 1.0$, (d) $\alpha_\theta = 0.75$ and $\alpha_{\mathbf{u}} = 1.0$, (e) $\alpha_\theta = 0.5$ and $\alpha_{\mathbf{u}} = 1.0$, and (f) $\alpha_\theta = 0.0$ and $\alpha_{\mathbf{u}} = 1.0$.

(continued from previous page)

```
# close output file
#
tOutFile.close();
```

The `opt_criteria_history.csv` file is written by the the topology optimization algorithm. It records the objective and constraint(s) function evaluation histories.

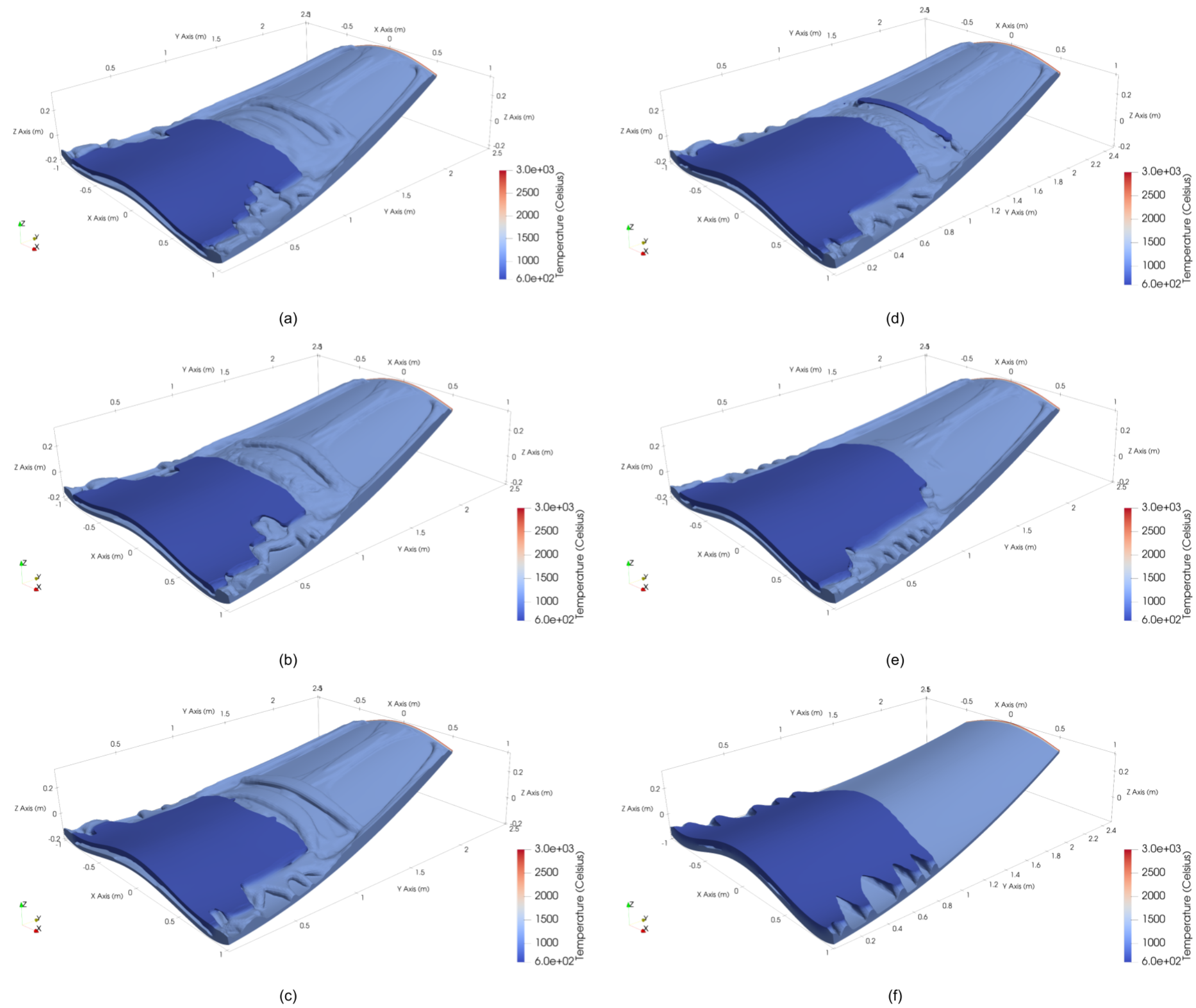

Fig. 3.4.8: Temperature field for the topology optimized structures. Only six out of the twenty-five samples are shown in Fig. 3.4.8. The plots correspond to the following samples: (a) $(\alpha_\theta = 1.0, \alpha_{\mathbf{u}} = 0.5)$, (b) $(\alpha_\theta = 1.0, \alpha_{\mathbf{u}} = 0.75)$, (c) $(\alpha_\theta = 1.0, \alpha_{\mathbf{u}} = 1.0)$, (d) $(\alpha_\theta = 0.75, \alpha_{\mathbf{u}} = 1.0)$, (e) $(\alpha_\theta = 0.5, \alpha_{\mathbf{u}} = 1.0)$, and (f) $(\alpha_\theta = 0.0, \alpha_{\mathbf{u}} = 1.0)$.

Run Study

To run the multidimensional parameter study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_tmech_ex2.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_tmech_ex2.i` is the *input deck* for the multidimensional parameter study. The design study should run after pressing the `Enter` key.

Results

Fig. 3.4.7 and Fig. 3.4.8 show the displacement and temperature field for the optimized structure. The structural stiffness increased as the thermal compliance objective was deprioritized and the weight of the mechanical compliance objective was kept at 1.0, i.e., $\alpha_{\mathbf{u}} = 1.0$. Therefore, design Fig. 3.4.7-f had the best mechanical performance (stiffess design) and the worst thermal performance. Likewise, the thermal performance improved as the mechanical compliance objective was deprioritized and the weight of the thermal compliance objective was kept at 1.0, i.e., $\alpha_\theta = 1.0$. Therefore, design Fig. 3.4.7-a had the best thermal performance and the worst mechanical performance (largest peak displacement). All the optimized design samples met the 70 percent mass reduction target.

3.4.3 Example 3: Local Stress Constraints

This thermomechanical topology optimization study seeks to minimize the structural mass and the peak displacement while satisfying a von Mises stress limit at each material point. This type of topology optimization problems become computationally intractable if the stress limit at each material point is not properly modeled. Indeed, if a gradient-baased optimization algorithm is used to solve the topology optimization proobelm, $1 + N_g$ large system of equations must be solved at each major optimization iteration k to compute the gradient of the objective function with respect to the design parameters. Here, N_g denotes the number of nonlinear constraints, i.e., stress constraint at each material point. Thus, an augmented Lagrangian approach is employed to efficiently solve optimization problems with a large number of nonlinear constraints.

Model Definition

The computational model used for this topology optimization study is shown in Fig. 3.4.1. The design domain is symmetric about the plane at $X = 0$ and is made of a linear elastic material. Thus, only half of the computational model (Fig. 3.4.1) is considered for topology optimization.

The structure is optimized for thermal and mechanical environments. The boundary conditions for the topology optimization study are:

1. Dirichlet boundary conditions
   a. The X, Y, and Z displacements (m) on the surface defined by the plane at $Y = 2.5$ are fixed.
   b. The X displacements on the symmetry surface defined by the plane at $X = 0$ are fixed.
   c. The Y displacements along the edge with starting point $(1, 0, -0.1667)$ and ending point $(0.5, 2.5, 0.2083)$ are fixed.
   d. The top surface is kept at $700°$C.
   e. The bottom surface is kept at $1200°$C.
   f. The front surface ddefined by the plane at $Y = 2.5$ is kept at $3000°$C.
2. External forces
   a. A concentrated force (N) with components $(0.0, 0.0, -6200)$ is applied on the top red surface.
   b. A concentrated force with components $(0.0, 0.0, -3475)$ is applied on the top blue surface.
3. Non-optimizable regions
   a. The nodes on the bottom surface are set to non-optimizable nodes.
   b. The nodes on the top red and blue surfaces are set to non-optimizable nodes.

The elastic modulus is set to 120 GPa and Poisson's ratio is set to 0.34. The material conductivity constant is set to 17 $\mathrm{W/m^2} \cdot^\circ$ C and the coefficient of thermal expansion is set to $1e^{-8}$ $\mathrm{m/m} \cdot^\circ$ C.

Formulation

The topology optimization problem seeks to achieve the following design goals:

- minimize the structural mass,
- minimize the peak displacement,
- improve heat transfer performance, and
- satisfy the von Mises stress limit at every material point.

Mathematically, this topology optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) = 0 \\ & g_j\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) \leq 0,\ j = 1, \ldots, N_g \\ & h(\mathbf{z}) \leq 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1, \ldots, N_z, \end{aligned} \tag{3.4.2}$$

where $N_f = 2$ is the number of objective functions and N_g is the number of nonlinear constraints. The state variables $\mathbf{u}(\mathbf{z})$ and $\theta(\mathbf{z})$ are the displacement vector and temperature, respectively. These state variables implicitly depend on the vector of design variables $\mathbf{z}$ of size N_z. $\mathbf{R}(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z})$ is the residual vector for the steady state thermomechanical physics. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$,respectively.

The first objective function $f_{l=1}$ is defined by the mechanical compliance criterion:

$$f_l\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) = \int_\Omega \sigma_{pq}\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) \epsilon_{pq}\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right)\ d\Omega \tag{3.4.3}$$

where Ω is the design domain. σ_{pq} is the stress tensor, where $p, q = 1, \ldots, d$, with $d \in \{1, 2, 3\}$ denoting the spatial dimension. ϵ_{pq} in Eq.3.4.3 is the strain tensor. The second objective function $f_{l=2}$ is defined by the thermal compliance criterion:

$$f_l\left(\theta(\mathbf{z}), \mathbf{z}\right) = \int_\Omega \kappa(\mathbf{z})\theta^2\ d\Omega \tag{3.4.4}$$

where $\kappa > 0$ is the material thermal conductivity constant. Each objective function is weighted by the scalar α_l, where $l = 1, \ldots, N_f$. The linear constraint $h(\mathbf{z})$ is defined by the volume criterion:

$$h\left(\mathbf{z}\right) = \int_\Omega \mathbf{z}\, d\Omega \tag{3.4.5}$$

The topology optimization problem in Eq.3.4.2 becomes computationally intractable if the nonlinear constraints g_j are not properly managed. An augmented Lagrangian approach is employed to properly model the stress constraints at each material point. Mathematically, the topology optimization problem in Eq.3.4.2 is recast as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_\mathbf{z}}} \quad & \sum_{l=1}^{N_f} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) + \frac{\beta}{N_g}\sum_{j=1}^{N_g}\left[\gamma_j^{(k)}\hat{g}_j\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) + \frac{\mu_j^{(k)}}{2}\hat{g}_j\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right)^2\right] \\ \text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) = 0 \\ & \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1, \ldots, N_z, \end{aligned} \tag{3.4.6}$$

where $\beta > 0$ and

$$\hat{g}_j\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) = \left(g_j\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right), -\frac{\gamma_j^{(k)}}{\mu_j^{(k)}}\right), \tag{3.4.7}$$

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} + \mu_j^{(k)} \hat{g}_j \left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right), \tag{3.4.8}$$

$$\mu_j^{(k+1)} = \left(\alpha \mu_j^{(k)}, \mu_{max}\right), \ \alpha > 1. \tag{3.4.9}$$

The augmented Lagrangian problem in Eq.3.4.6 is solved until a convergence criteria is satisfied. The parameters $\gamma_j^{(k)}$ and $\mu_j^{(k)}$ are the j-th Lagrange multiplier and penalties, respectively. These two parameters are related to the j-th nonlinear constraints $g_j^{(k)}$, i.e., the stress limit.

Input Deck

The input deck setup for the topology optimization study with local von Mises stress constraints is:

Listing 3.4.2: Input deck setup for the topology optimization study.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
  update_problem true
  update_problem_frequency 10
end service

begin criterion 1
  type thermomechanical_compliance
  thermal_weighting_factor 1.0
  mechanical_weighting_factor 1.0
end criterion

begin criterion 2
  type volume
end criterion

begin criterion 3
  type local_constraint
  limits 1e8
  local_measures vonmises
  initial_penalty 0.1
  minimum_ersatz_material_value 1e-6
end criterion

begin scenario 1
  service 2
  dimensions 3
  physics steady_state_thermomechanics
  loads 1 2
  boundary_conditions 1 2 3 4 5 6
```

(continues on next page)

(continued from previous page)

```
  linear_solver 1
  newton_raphson 1
  blocks 1
  output 1
end scenario

begin newton_raphson 1
  residual thermomechanical
  residual_tolerance 1e-8
  increment_tolerance 1e-8
  max_iterations 100
  num_increments 1
end newton_raphson

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 1 3
  services 2 2
  scenarios 1 1
  weights 1e-6 1e3
end objective

begin constraint 1
  criterion 2
  relative_target 0.3
  type less_than
  service 2
  scenario 1
end constraint

begin output 1
  service 2
  output_frequency 20
  native_service_output true
  data dispx dispy dispz temperature stress vonmises thermal_flux
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
```

(continues on next page)

(continued from previous page)

```
  location_name symm_plane
  degree_of_freedom dispx
  value 0.
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
  location_name le
  degree_of_freedom dispz
  value 0.
end boundary_condition

begin boundary_condition 4
  type fixed_value
  location_type nodeset
  location_name lower_surf
  degree_of_freedom temp
  value 1200.
end boundary_condition

begin boundary_condition 5
  type fixed_value
  location_type nodeset
  location_name front_surf
  degree_of_freedom temp
  value 3000.
end boundary_condition

begin boundary_condition 6
  type fixed_value
  location_type nodeset
  location_name top_surf
  degree_of_freedom temp
  value 700.
end boundary_condition

begin load 1
  type concentrated_load
  location_type nodeset
  location_name actuator1
  direction z
  value -6200
end load

begin load 2
  type concentrated_load
  location_type nodeset
  location_name actuator2
  direction z
  value -3475
end load
```

(continues on next page)

(continued from previous page)

```
begin block 1
  material 1
  name design_area
end block

begin material 1
  material_model isotropic_linear_thermoelastic
  poissons_ratio 0.34
  youngs_modulus 120e9
  thermal_expansivity 1e-8
  thermal_conductivity 17.
  reference_temperature 0.
end material

begin study 1
  method topology
  initialize_method uniform
  initial_density_value 1.0

  optimization_algorithm mma
  max_iterations 200
  mma_move_limit 0.1
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 1.75
  fixed_exodus_sideset_names lower_surf_ss
  fixed_exodus_nodeset_names actuator1 actuator2
end study

begin mesh
  name topology_optimization_tmech_ex3.exo
end mesh
```

Run Study

To run the topology optimization study with local von Mises stress constraints from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_tmech_ex3.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_tmech_ex3.i` is the *input deck* for the topology optimization study. The design study should run after pressing the `Enter` key.

Results

The thermal and mechanical compliance objectives were prioritized equally in the topology optimization study, i.e., $\alpha_1 = \alpha_2 = 1.0 \times 10^{-6}$. The weight β was set to 1.0×10^3, which is used to weight the augmented Lagrangian term in Eq.3.2.6. Fig. 3.4.9 shows the displacement and the von Mises stress fields for the topology optimized structure. The topology optimizer put most of the material near the loaded areas, i.e., areas were the concentrated forces were applied. In contrast, the optimizer proceeded to remove material in areas located far away from the loaded areas. The results in Fig. 3.4.9 show that the von Mises stress limit was met at most locations, with the exception of regions near the loaded

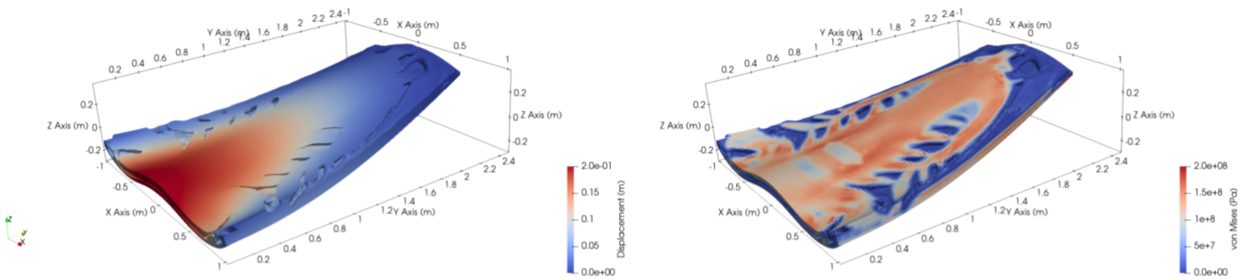

Fig. 3.4.9: The displacement (left pane) and von Mises stress (right pane) fields for the topology optimized structure.

areas. The peak displacement was measured at 0.211 m.

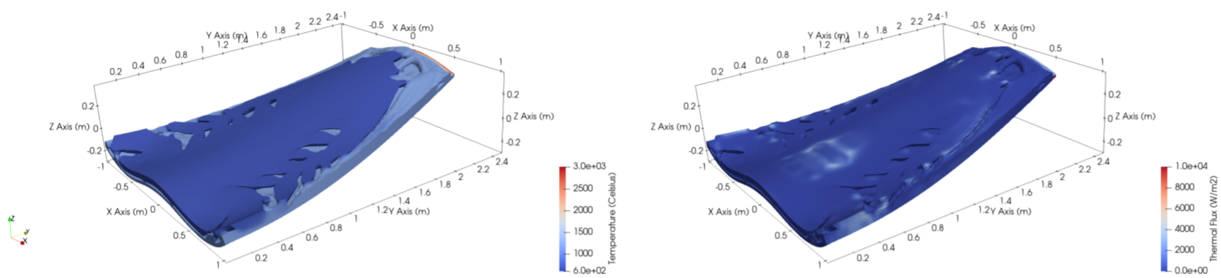

Fig. 3.4.10: The temperature (left pane) and thermal flux (right pane) fields for the topology optimized structure.

Fig. 3.4.10 shows the temperature and the thermal flux fields for the topology optimized structure. The topology optimizer proceeded to add sophisticated features near the high-temperature regions (front surface defined by the plane at $Y = 2.5$) to maximize the heat transfer performance. The temperature at the back surface defined by the plane at $Y = 0$ decreased from 700°C to 694°C.

3.4.4 Example 4: Multi-Scenario Optimization

This thermomechanical topology optimization study seeks to improve the heat transfer performance of a structural component while decreasing the peak displacement. A mass budget is given and the structural component is design for thermal and mechanical environments. This design setup is labeled multi-environment (multi-load) design optimization problem in this manuscript. The topology optimization algorithm employs a Multiple-Program, Multiple-Data (MPMD) parallel framework to concurrently evaluate the design environments. The MPMD framework requires access to the computational resources needed to perform the concurrent evaluations. If the computational resources are limited, the multi-scenario design optimization framework is available to enable the optimizer to sequentially evaluate the design environments. The purpose of *Example 4: Multi-Scenario Optimization* is to show how to setup a multi-scenario topology optimization problem.

Model Definition

The computational model, the finite element mesh, and the locations where boundary conditions are applied are shown in Fig. 3.4.11. The design domain is made of a linear elastic material. The structural component is optimized for thermal and mechanical environments. The boundary conditions for the topology optimization study are:

1. Dirichlet boundary conditions

    a. The purple surfaces shown in the right pane of Fig. 3.4.11 are kept at 311 K.

    b. The X, Y, and Z displacements m at the purple surfaces are fixed.

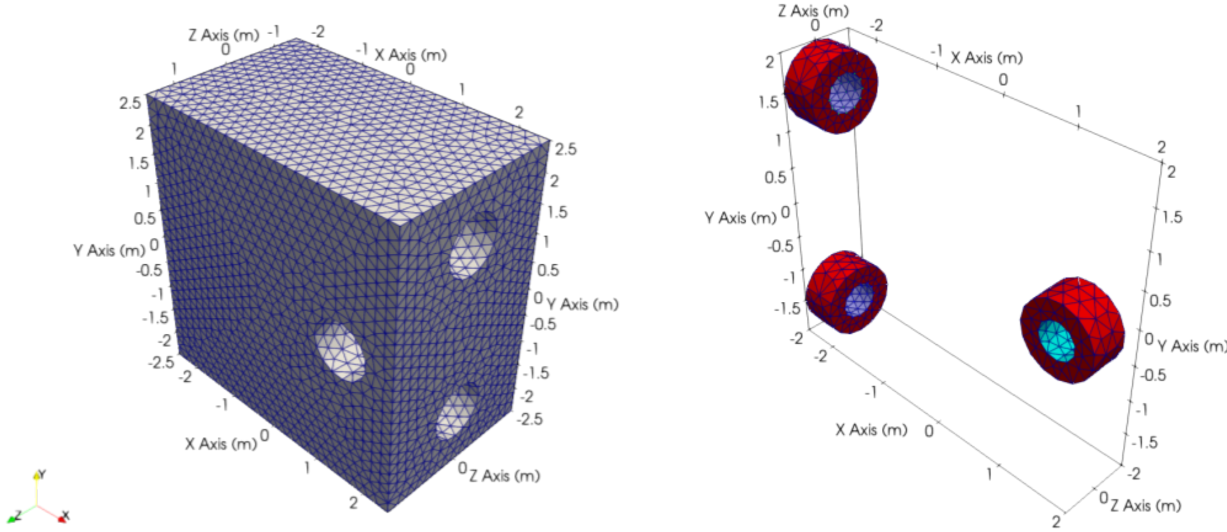

Fig. 3.4.11: The left pane shows the computational domain and the finite element mesh used for the thermomechancal topology optimization study. The right pane shows the locations where boundary conditions are applied and the non-optimizable regions. The purple surfaces shown in the right pane are kept at 311 K. The X, Y, and Z displacements (m) at the purple surfaces are fixed. The X, Y, and Z displacements on the red surfaces defined by the plane at $X = -2.5$ are fixed. The red material regions shown in the right pane are considered non-optimizable regions. Therefore, the optimizer cannot add nor removed material at these locations.

   c. The X, Y, and Z displacements at the red surface defined by the plane at $X = -2.5$ are fixed.

2. Neumann boundary conditions

   a. A thermal flux $(\mathrm{W/m^2})$ of -5000 is applied at the cyan surface shown in the right pane of Fig. 3.4.11.

   b. A traction force (Pa) with components $(0.0, -10000, 0.0)$ is applied at the cyan surface shown in the right pane of Fig. 3.4.11.

The elastic modulus is set to 68 GPa and Poisson's ratio is set to 0.32. The material conductivity constant is set to $220\ \mathrm{W/m^2 \cdot K}$ and the coefficient of thermal expansion is set to $24 \times 10^{-6}\ \mathrm{m/m \cdot K}$. The reference temperature is set to 0.0 K.

Formulation

The topology optimization problem seeks to reduce the structural mass and the peak displacement while improving the heat transfer performance of the structural component. Mathematically, this topology optimization problem is defined as:

$$
\begin{aligned}
\min_{\mathbf{z}\in\mathbb{R}^{N_{\mathbf{z}}}} \quad & \sum_{l=1}^{N_s} \alpha_l f_l\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) \\
\text{s.t.} \quad & \mathbf{R}\left(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z}\right) = 0 \\
& h(\mathbf{z}) \leq 0 \\
& \underline{z}_i \leq z_i \leq \bar{z}_i,\ i = 1, \ldots, N_z,
\end{aligned}
\tag{3.4.10}
$$

where $N_s = 2$ is the number of scenarios. The state variables $\mathbf{u}(\mathbf{z})$ and $\theta(\mathbf{z})$ are the displacement vector and temperature, respectively. The state variables implicitly depend on the vector of design variables $\mathbf{z}$ of size N_z. $\mathbf{R}(\mathbf{u}(\mathbf{z}), \theta(\mathbf{z}), \mathbf{z})$ is the residual vector for the steady state thermomechanical physics. The lower and upper bounds for the i-th design variable z_i are given by $\underline{z}_i$ and $\bar{z}_i$, respectively.

The objective function is defined by the thermomechanical compliance criterion:

$$f_l\left(\mathbf{u}(\mathbf{z}),\theta(\mathbf{z}),\mathbf{z}\right)=\int_{\Omega}\beta_l^u\sigma_{pq}\left(\mathbf{u}(\mathbf{z}),\theta(\mathbf{z}),\mathbf{z}\right)\epsilon_{pq}\left(\mathbf{u}(\mathbf{z}),\theta(\mathbf{z}),\mathbf{z}\right)\ d\Omega+\int_{\Omega}\beta_l^{\theta}\kappa(\mathbf{z})\theta^2\ d\Omega \tag{3.4.11}$$

where $\beta_l^u>0$ and $\beta_l^{\theta}>0$ are the l-th mechanical and thermal compliance weights, respectively. The design domain is denoted with the parameter Ω. σ_{pq} is the stress tensor, where $p,q=1,\dots,d$, with $d\in\{1,2,3\}$ denoting the spatial dimension. ϵ_{pq} is the strain tensor and $\kappa>0$ is the material thermal conductivity constant. Each scenario is weighted by a positive scalar $\alpha_l=1.0$, where $l=1,\dots,N_s$ denotes the scenario index. Finally, the constraint $h(\mathbf{z})$ is defined by the volume criterion:

$$h\left(\mathbf{z}\right)=\int_{\Omega}\mathbf{z}\,d\Omega \tag{3.4.12}$$

Input Deck

The input deck setup for the multi-scenario topology optimization study is:

Listing 3.4.3: Input deck setup for the multi-scenario topology optimization study.

```
begin service 1
  code engine
  number_processors 1
  number_ranks 1
end service

begin service 2
  code analyze
  number_processors 1
  number_ranks 1
end service

begin criterion 1
  type thermomechanical_compliance
  thermal_weighting_factor 1
  mechanical_weighting_factor 0
end criterion

begin criterion 2
  type thermomechanical_compliance
  thermal_weighting_factor 1
  mechanical_weighting_factor 1e-1
end criterion

begin criterion 3
  type volume
end criterion

begin scenario 1
  dimensions 3
  physics steady_state_thermomechanics
  service 2
  material 1
  blocks 1 2
```

(continues on next page)

(continued from previous page)

```
  loads 1
  output 1
  boundary_conditions 1 2 3 4 5
  minimum_ersatz_material_value 1e-9
end scenario

begin scenario 2
  dimensions 3
  physics steady_state_thermomechanics
  service 2
  material 1
  blocks 1 2
  loads 1 2
  output 1
  boundary_conditions 1 2 3 4 5
  minimum_ersatz_material_value 1e-9
end scenario

begin objective
  type weighted_scenarios
  scenarios 1 2
  services 2 2
  criteria 1 2
  weights 1 1
end objective

begin constraint 1
  criterion 3
  relative_target 0.3
  type less_than
  service 1
end constraint

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name ns_1
  degree_of_freedom temp
  value 311
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom temp
  value 311
end boundary_condition

begin boundary_condition 3
  type fixed_value
  location_type nodeset
```

(continues on next page)

(continued from previous page)

```
  location_name ns_1
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 4
  type fixed_value
  location_type nodeset
  location_name ns_2
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin boundary_condition 5
  type fixed_value
  location_type sideset
  location_name ss_3
  degree_of_freedom dispx dispy dispz
  value 0. 0. 0.
end boundary_condition

begin load 1
  type thermal_flux
  location_type sideset
  location_name ss_1
  value -5e4
end load

begin load 2
  type traction
  location_type sideset
  location_name ss_1
  value 0 -1e4 0
end load

begin block 1
  material 1
  location_name block_1
end block

begin block 2
  material 1
  location_name block_2
end block

begin material 1
  material_model isotropic_linear_thermoelastic
  thermal_conductivity 220
  poissons_ratio 0.32
  youngs_modulus 68e9
  reference_temperature 0
  thermal_expansivity 24e-6
```

(continues on next page)

(continued from previous page)

```
end material

begin output 1
  service 2
  output_frequency 10
  native_service_output true
  data temperature dispx dispy dispz vonmises thermal_flux
end output

begin study
  method topology
  discretization density
  initial_density_value 1.0

  max_iterations 100
  mma_move_limit 0.5
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true

  filter_radius_scale 2
  fixed_block_ids 2
end study

begin mesh
  name topology_optimization_tmech_ex4.exo
end mesh
```

Run Study

To run the multi-scenario topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_tmech_ex4.i
```

runmorphorm.py is the python script holding the sequence of instructions to run the Morphorm software and topology_optimization_tmech_ex4.i is the *input deck* for the multi-scenario topology optimization study. The design study should run after pressing the Enter key.

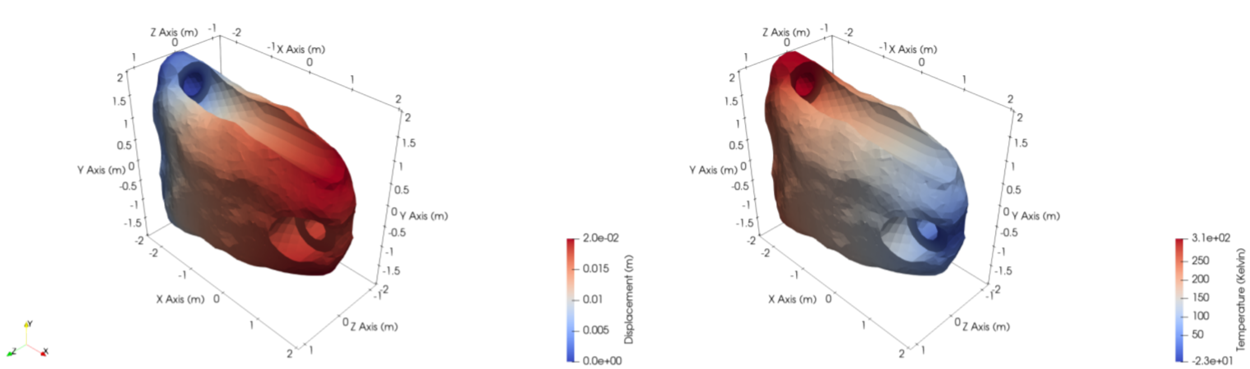

Fig. 3.4.12: The displacement (right pane) and temperature (left pane) fields for the optimized configuration.

Results

The thermal and mechanical compliance objectives were given different prioritization based on the scenario. Table 3.4.1 shows the numerical values assigned to the mechanical (β_l^u) and thermal (β_l^θ) compliance weights. Fig. 3.4.12 shows the displcement and temperature field for the optimized bracket.

Table 3.4.1: Numerical values assigned to the mechanical (β_l^u) and thermal (β_l^θ) compliance weights based on the scenario.

Scenario	Thermal Compliance Weight	Mechanical Compliance Weight
$l = 1$	1.0	0.0
$l = 2$	1.0	0.1

3.5 Nonlinear Mechanics

The following topology optimization examples demonstrate how to use density-based topology optimization to optimize the distribution of a fixed amount of material such that the mechanical design criteria are improved. The examples create a suite of tutorials which summarize the main concepts for solving mechanical topology optimization problems modeling geometric nonlinearities.

3.5.1 Example 1: Maximize Deformations

This nonlinear mechanical topology optimization problem seeks to find a lightweight frame such that the peak displacement is maximized. The optimizer must balance several conflicting design criteria: peak displacement maximization and mass.

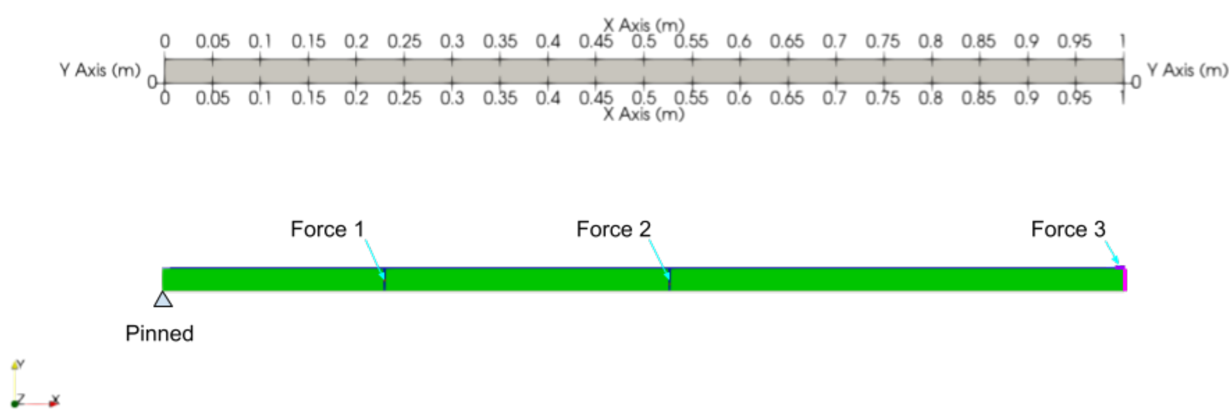

Fig. 3.5.1: The top pane shows the computational domain for the topology optimization study. The bottom pane shows the locations where boundary conditions are applied. The X displacements (m) on the symmetry plane defined by the plane at $X = 1.0$ are fixed. The Y displacement at point $(0.0, 0.0)$ is fixed. Three actuating forces (N) are applied at the locations depicted in the figure with the `Force 1`, `Force 2`, and `Force 3` identifiers.

Model Definition

Fig. 3.5.1 shows the computational domain and the boundary condition setup used for the topology optimization study. The design domain is symmetric about the plane $X = 1.0$ and is made of a linear elastic material. The design is optimized for mechanical loads. The boundary conditions for the topology optimization study are:

1. Dirichlet boundary conditions

   a. The Y displacement (m) at point $(0., 0.)$ is fixed.

   b. The X displacements on the surface defined by the plane at $X = 1.0$ are fixed.

2. External forces

   a. A concentrated force (N) with components $(0.0, -5.0 \times 10^2)$ is applied at point $(0.231, 0.012)$.

   b. A concentrated force with components $(0.0, -1.0 \times 10^3)$ is applied at point $(0.527, 0.008)$.

   c. A concentrated force with components $(0.0, -2.5 \times 10^3)$ is applied at point $(1.0, 0.25)$.

3. Non-optimizable regions

   a. The nodes at the top and bottom surfaces are set to non-optimizable regions.

   b. The nodes at the vertical blue strips where the forces are applied are set to non-optimizable regions.

The shear modulus is set to 122.18 GPa and the bulk modulus is set to 318.63 GPa.

Formulation

Given a mass budget, the optimization problem seeks to find a flexible and lightweight frame to maximize the peak displacement. Mathematically, the topology optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{z}\in\mathbb{R}^{N_z}} \quad & \alpha f(\mathbf{u}(\mathbf{z}),\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z}) = 0 \\ & h(\mathbf{z}) \leq 0 \\ & \underaccent{\bar}{z}_i \leq z_i \leq \bar{z}_i,\ i = 1,\ldots,N_z \end{aligned} \tag{3.5.1}$$

where $\mathbf{u}(\mathbf{z})$ is the displacement vector, which depends on the vector of design variables $\mathbf{z}$ of size N_z. $f(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the internal elastic energy criterion, which is defined as:

$$f(\mathbf{u}(\mathbf{z}),\mathbf{z}) = \int_\Omega k_{pq}(\mathbf{z}) u_p u_q \, d\Omega \tag{3.5.2}$$

where Ω is the design domain, $k_{pq}(\mathbf{z})$ is the element stiffness matrix, u_p is the p-th displacement, and $p, q = 1, \ldots, d$, with $d \in \{1, 2, 3\}$ denoting the spatial dimension. The scalar weight α is set to -1.0, i.e., maximize the objective function. Maximizing the internal elastic energy will maximize the peak deformation. $\mathbf{R}(\mathbf{u}(\mathbf{z}),\mathbf{z})$ is the residual vector for the nonlinear mechanical physics and $h(\mathbf{z})$ is the mass budget constraint. The lower and upper bounds for the i-th design variable z_i are given by $\underaccent{\bar}{z}_i$ and $\bar{z}_i$, respectively. Finally, a minimum feature size is introduced via the length scale parameter of the Kernel filter. The method of moving asymptotes is used to solve Eq.3.5.1.

Input Deck

The input deck setup for the nonlinear mechanical topology optimization study is:

Listing 3.5.1: Input deck setup for the topology optimization study.

```
begin service 1
  code engine
  number_processors 1
  update_problem true
  update_problem_frequency 1
end service

begin service 2
  code analyze
  number_processors 1
  update_problem true
```

(continues on next page)

(continued from previous page)

```
  update_problem_frequency 1
end service

begin criterion 1
  type elastic_energy_potential
  minimum_ersatz_material_value 1e-3
end criterion

begin criterion 2
  type volume
end criterion

begin scenario 1
  physics steady_state_nonlinear_mechanics
  service 2
  dimensions 2
  material 1
  loads 1 2 3
  blocks 1 2 3 4
  boundary_conditions 1 2
  newton_raphson 1
  linear_solver 1
  output 1
  material_penalty_exponent 2.0
  minimum_ersatz_material_value 1e-3
end scenario

begin newton_raphson 1
  residual mechanical
  residual_tolerance 1e-8
  increment_tolerance 1e-8
  max_iterations 100
  num_increments 20
end newton_raphson

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 1
  services 2
  scenarios 1
  weights -1
end objective

begin constraint 1
  criterion 2
  relative_target 0.5
  type less_than
  service 2
```

(continues on next page)

(continued from previous page)

```
  scenario 1
end constraint

begin output 1
  service 2
  output_frequency 1
  data dispx dispy vonmises
  native_service_output true
end output

begin boundary_condition 1
  type fixed_value
  location_type nodeset
  location_name pin_node
  degree_of_freedom dispy
  value 0.
end boundary_condition

begin boundary_condition 2
  type fixed_value
  location_type nodeset
  location_name symm_nodes
  degree_of_freedom dispx
  value 0.
end boundary_condition

begin load 1
  type concentrated_load
  location_type nodeset
  location_name p1_node
  direction x y
  value 0 -1e3
  index 0
end load

begin load 2
  type concentrated_load
  location_type nodeset
  location_name p2_node
  direction x y
  value 0 -5e2
  index 0
end load

begin load 3
  type concentrated_load
  location_type nodeset
  location_name p3_node
  direction x y
  value 0 -2.5e3
end load
```

(continues on next page)

(continued from previous page)

```
begin block 1
  material 1
  name frozen_area
  initial_dvars_value 1.0
end block

begin block 3
  material 1
  name design_area
  initial_dvars_value 1.0
end block

begin block 2
  material 1
  name actuator_pad
  initial_dvars_value 1.0
end block

begin block 4
  material 1
  name init_density_area
  initial_dvars_value 1.0
end block

begin material 1
  material_model isotropic_hyperelastic_neohookean
  shear_modulus 122.18e9
  bulk_modulus 318.63e9
  penalty_exponent 2.0
  minimum_ersatz_material_value 1e-3
end material

begin study
  method topology
  discretization density
  filter_radius_scale 2.0
  initialize_method uniform_by_region

  projection_type tanh
  filter_heaviside_max 50
  filter_projection_start_iteration 50
  filter_projection_update_interval 5
  filter_heaviside_update 2
  filter_use_additive_continuation true

  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true
  mma_move_limit 0.1
  max_iterations 100

  fixed_block_ids 1 2 4
end study
```

(continues on next page)

(continued from previous page)

```
begin mesh
  name topology_optimization_nlmech_ex1.exo
end mesh
```

The `initialize_method` keyword within the `study block` is set to `uniform_by_region` for this example. The `uniform_by_region` initialization scheme enables the user to initialize the design variables by material regions, i.e., `exodus element blocks`. The `initial_dvars_value` keyword within the `block block` is used in combination with the `initialize_method` keyword to set a uniform value to the design variables assigned to this region.

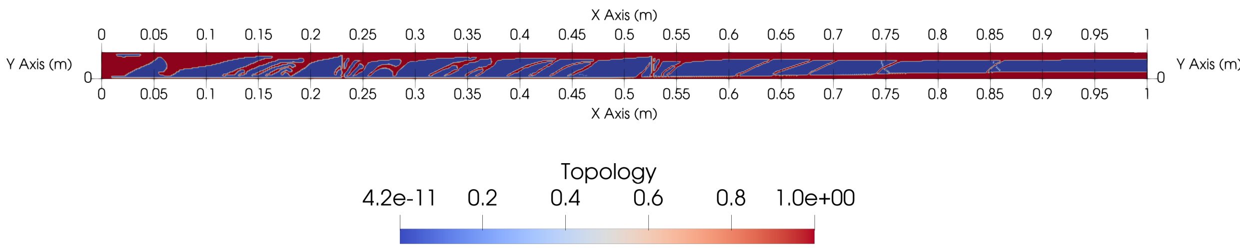

Fig. 3.5.2: The density field for the topology optimized frame.

Fig. 3.5.3: The computational model used for the verification study. The boundary conditions for the verification study are similar to the boundary conditions used for the topology optimization study.

The `prune and refine` application is used to generate a computational model from the density field shown in Fig. 3.5.2, which is the solution to the first topology optimization study. The pruned computational model shown in Fig. 3.5.3 is used to run a verification study. The input deck setup for the `prune and refine` application is:

Listing 3.5.2: Input deck setup for the prune and refine application.

```
begin service 1
  code prune
  number_processors 1
end service

begin prune
  prune_threshold 0.1
  number_of_refines 1
  spatial_dimensions 2
  number_buffer_layers 1
  initial_guess_field_name Topology
  initial_guess_mesh iteration100.exo
end prune

begin study
  method prune
end study

begin mesh
```

(continues on next page)

(continued from previous page)

```
  name topology_optimization_nlmech_ex1.exo
end mesh
```

The `iteration100.exo` file is written to the `analyze_id_2_output_scenario_id_1` directory. This directory holds the output information for the topology optimization study. The output files inside the `analyze_id_2_output_scenario_id_1` directory are written by the simulation toolkit, i.e., `Analyze`.

The output from the simulation toolkit is disabled by default for optimization studies. To enable the output from the simulation toolkit, the `native_service_output` keyword within the `output block` must be set to `true`. The rate at which output information is written to disk by `Analyze` for design optimization studies can be adjusted using the `output_frequency` keyword. The output rate for optimization studies is set to 1 by default. Therefore, the simulation toolkit will write the output quantities of interests (e.g., displacements) for the optimized configuration at each major optimization iteration.

Run Study

To run the nonlinear mechanical topology optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py topology_optimization_nlmech_ex1.i
```

where `runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `topology_optimization_nlmech_ex.i` is the *input deck* for the nonlinear mechanical topology optimization study. The design study should run after pressing the `Enter` key.

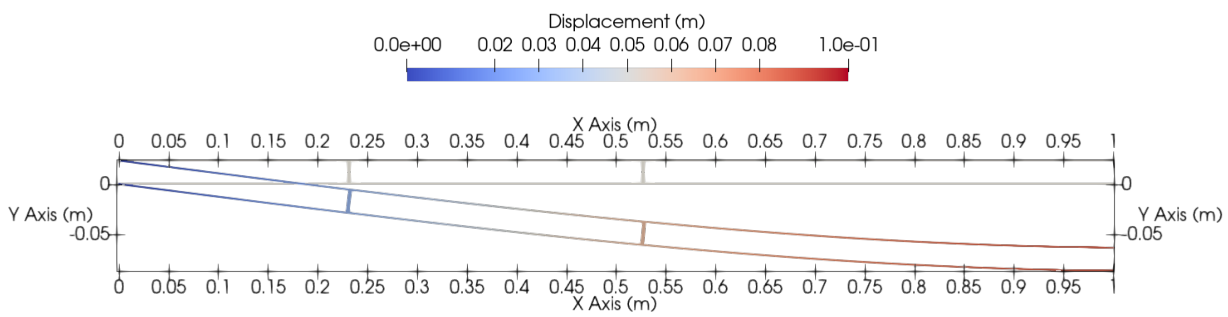

Fig. 3.5.4: The undeformed topology optimized frame and the corresponding displacement field plotted on the deformed topology optimized frame. The optimized framed is the solution to the topology optimization study modeling linear mechanical physics.

Results

Fig. 3.5.4 shows the displacement field for the optimized structural frame obtained using topology optimization methods modeling linear mechanical physics. The boundary conditions and external forces used for the linear problem were the same as the ones used for the *nonlinear problem*. In the linear case, the topology optimizer produced the trivial solution, which is to remove the internal material to maximize the peak displacement, i.e., $u_{\max} = 0.08712$ m at point $(1.0, 0.25)$.

Fig. 3.5.5 shows the displacement field for the optimized structural frame obtained using topology optimization methods modeling nonlinear mechanical physics. In contrast to the linear case, the topology optimizer modeling nonlinear physics proceeded to add structural members to support the internal forces generated by the large deformations and rotations. The peak displament measured for the optimized frame obtained with the nonlinear topology optimization approach was measured at point $(1.0, 0.0005)$ to be $u_{\max} = 0.0421$ m.

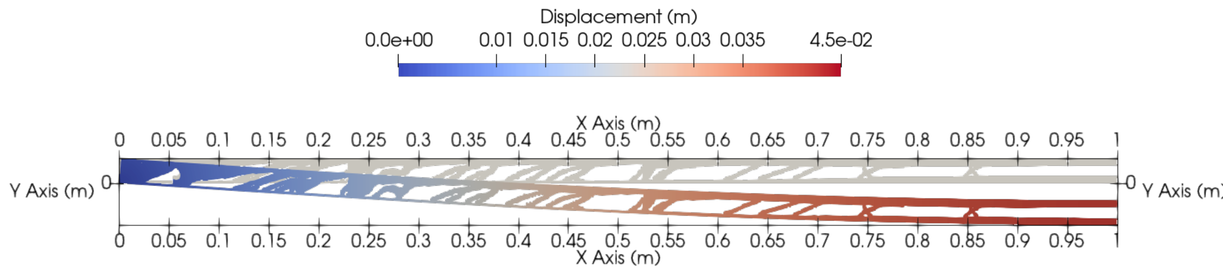

Fig. 3.5.5: The undeformed topology optimized frame and the corresponding displacement field plotted on the deformed topology optimized frame. The optimized framed is the solution to the topology optimization study modeling geometric nonlinearities.

The 50 percent mass reduction target was met for the topology optimization study modeling nonlinear mechanical physics. However, the topology optimization study modeling linear mechanical physics overshoots this design requirement since the optimizer will seek to remove as much material as possible to maximize the peak displacement, i.e., the trivial solution. Indeed, if the frozen domain constraints are removed, the topology optimizer will remove all the material from the design domain to maximize the peak displacement.

CHAPTER

FOUR

SHAPE OPTIMIZATION

The Shape Optimization chapter provides a suite of examples describing how to use the Optimize module to set up and run shape optimization problems.

4.1 Description

Shape optimization is a mathematical method used to optimize the parameters that define the computational geometry. Shape optimization methods operate on a restricted set of shape parameters, i.e., Computer-Aided Design parameters, that define the geometry. In contrast, topology optimization methods are free to select any material layout within the design space as long as the design goals are satisfied.

4.2 Example 1: Gradient-Based Optimization

This shape optimization problem aims to find a lightweight structure such that the stiffness of the structure is maximized. The optimizer must balance several conflicting design criteria: stiffness and volume.

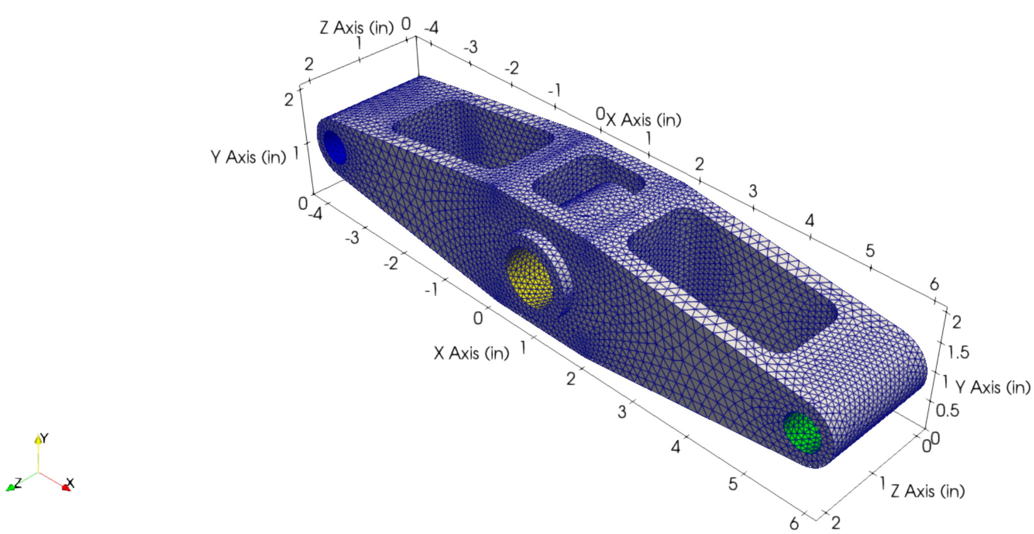

Fig. 4.2.1: The computational model and the finite element mesh used for the shape optimization study. The X, Y, and Z displacements (in) on the yellow surface along the center axis are fixed. A traction force (lbf) with components $(0.0, 450.0, 0.0)$ is applied on the blue surface located close to the plane at $X = 0$. A second traction force (lbf) with components $(0.0, 225.0, 0.0)$ is applied on the green surface located close to the plane at $X = 6.25$.

4.2.1 Model Definition

Fig. 4.2.1 shows the initial computational domain and the finite element mesh for the shape optimization study. The part is made of a linear elastic material. The boundary conditions for the shape optimization study are:

1. Dirichlet boundary conditions

   a. The X, Y, and Z displacements (in) on the yellow surface along the center axis are fixed.

2. Neumann boundary conditions

   a. A traction force (lbf) with components $(0.0, 450.0, 0.0)$ is applied on the blue surface, see Fig. 4.2.1.

   b. A traction force (lbf) with components $(0.0, 225.0, 0.0)$ is applied on the green surface, see Fig. 4.2.1.

The elastic modulus is set to 29007548 psi and Poisson's ratio is set to 0.33.

4.2.2 Formulation

The shape optimization problem seeks maximize the structural stiffness while satisfying a mass/volume budget. Mathematically, the shape optimization problem is defined as:

$$\begin{aligned} \min_{\mathbf{p}\in\mathbb{R}^{N_p}} \quad & \alpha f(\mathbf{u}(\mathbf{p}),\mathbf{p}) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{p}(\mathbf{z}),\mathbf{p}) = 0 \\ & h(\mathbf{p}) \leq 0 \\ & \underline{p}_i \leq p_i \leq \bar{p}_i,\ i = 1,\dots,N_p \end{aligned} \tag{4.2.1}$$

where the weight $\alpha = 1.0$. The displacement vector $\mathbf{u}(\mathbf{p})$ depends on the vector of shape parameters $\mathbf{p}$ of size N_p. The objective function $f(\mathbf{u}(\mathbf{p}),\mathbf{p})$ is defined by the mechanical compliance criterion:

$$f(\mathbf{u}(\mathbf{p}),\mathbf{p}) = \int_\Omega k_{pq} u_p u_q \, d\Omega(\mathbf{p}) \tag{4.2.2}$$

where $\Omega(\mathbf{p})$ is the design domain, k_{pq} is the element stiffness matrix, u_p and u_q are the p-th and q-th displacements, where $p, q = 1, \dots, d$. The parameter $d \in \{1, 2, 3\}$ denotes the spatial dimension. $\mathbf{R}(\mathbf{u}(\mathbf{p}),\mathbf{p})$ is the residual vector for the linear mechanical physics and $h(\mathbf{p})$ is the mass/volume constraint:

$$f(\mathbf{p}) = \int_\Omega d\Omega(\mathbf{p}) \tag{4.2.3}$$

The lower and upper bounds for the i-th shape parameter p_i are denoted with $\underline{p}_i$ and $\bar{p}_i$, respectively. The method of moving asymptotes is used to solve Eq.4.2.1.

4.2.3 Input Deck

The input deck setup for the shape optimization study is:

Listing 4.2.1: Input deck setup for the shape optimization study.

```
begin service 1
  code engine
  number_processors 1
end service

begin service 2
  code analyze
  number_processors 1
end service
```

(continues on next page)

(continued from previous page)

```
begin service 3
  code esp
  number_processors 5
end service

begin criterion 1
  type mechanical_compliance
end criterion

begin criterion 2
  type volume
  minimum_ersatz_material_value 0
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1
  material 1
end scenario

begin objective
  type weighted_sum
  criteria 1
  services 2
  shape_services 3
  scenarios 1
  weights 1
end objective

begin output
  service 2
  output_frequency 5
  native_service_output true
  data dispx dispy dispz
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name center_axis
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name left_axis
```

(continues on next page)

(continued from previous page)

```
  value 0 450 0
end load

begin load 2
  type traction
  location_type sideset
  location_name right_axis
  value 0 225 0
end load

begin constraint 1
  criterion 2
  absolute_target 17.5
  type less_than
  service 2
  scenario 1
end constraint

begin block 1
  material 1
end block

begin material 1
  material_model isotropic_linear_elastic
  poissons_ratio 0.33
  youngs_modulus 29007548
end material

begin study
  method shape
  max_iterations 25
  optimization_algorithm mma
  mma_use_ipopt_sub_problem_solver true
  normalize_in_aggregator false
  csm_file shape_optimization_mech_ex1.csm
  num_shape_design_variables 5
end study

begin mesh
  name shape_optimization_mech_ex1.exo
end mesh
```

The shape parameter sensitivity is composed of two parts. The first part is the sensitivity of the design criterion with respect to the configuration and the second part is the sensitivity of the configuration with respect to the shape parameters. The sensitivity of the design criterion with respect to the configuration is computed by the simulation toolkit, i.e., `Analyze`. The sensitivity of the configuration with respect to the shape parameters is computed by the `Engineering Sketch Pad (ESP)` application [Haimes and Dannenhoffer]. The shape optimizer combines these two sensitivities to compute the gradient of the design criteria with respect to the shape parameters. The `csm_file` keyword is used to define the `csm` file describing the parameterized configuration in ESP.

4.2.4 Run Study

To run the shape optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py shape_optimization_mech_ex1.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `shape_optimization_mech_ex1.i` is the *input deck* for the shape optimization study. The design study should run after pressing the `Enter` key.

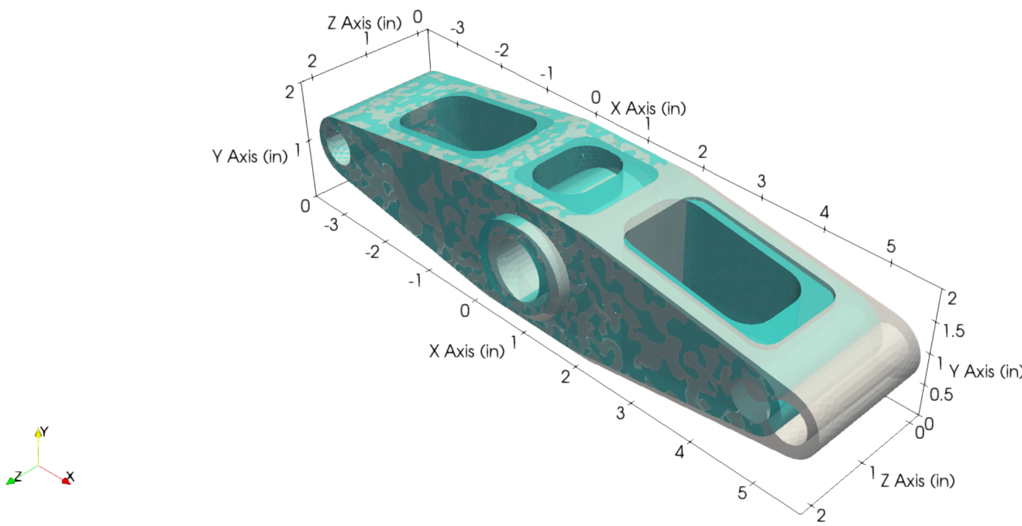

Fig. 4.2.2: The optimized configuration (cyan) overlaid with the baseline configuration (gray).

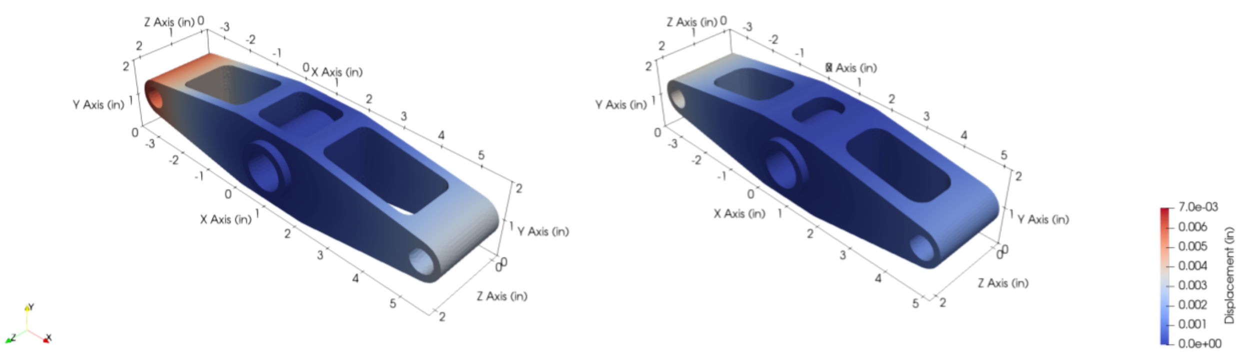

Fig. 4.2.3: The displacement field for the baseline (left pane) and optimized (right pane) configurations.

4.2.5 Results

Fig. 4.2.2 shows the optimized configuration overlaid with the baseline configuration. The volume of the optimized configuration was reduced from 17.5 in^3 to 16.03 in^3. Fig. 4.2.3 shows the displacement field for the baseline and optimized configurations. Results confirm that the peak displacement was reduced from 7.0×10^{-3} to 3.5×10^{-3}, which is a fifty percent reduction in peak displacement.

4.3 Example 2: Surrogate-Based Optimization

Listing 4.3.1: Input deck setup for the surrogate-based shape optimization study.

```
begin variables 1
  numvars 5
  type continuous_design
  initial_points 2.5 0.25 6.5 6.5 2.0
  lower_bounds   2.0 0.05 5.0 5.0 1.6
  upper_bounds   3.0 0.38 8.0 8.0 2.4
  descriptors     p1   p2  p3  p4  p5
end variables

begin study
  method surrogate_based_global_optimization
  variables 1
  num_criteria 1
  num_samples 8
  max_iterations 50
  concurrent_evaluations 4
  copy_files analysis_driver.sh runmorphorm.py mech_analysis_ex2.i postprocess.py esp_
↪geometry_file.temp
end study
```

This shape optimization study aims to reduce the peak displacement (maximize stiffness) given a mass budget. The previous example, *Example 1: Gradient-Based Optimization*, used a gradient-based shape optimization method to find the shape parameters that improved the performance of the structural component in Fig. 4.2.1. In contrast to the previous example, a surrogate-based shape optimization algorithm is used in this example to enable fast exploration of the design space. The initital computational model and the boundary conditions remain the same as those used in *Example 1: Gradient-Based Optimization*. The shape optimization formulation is also similar to the formulation introduced in *Example 1: Gradient-Based Optimization*, section *Formulation*. However, a Bayesian optimization framework is used to solve the shape optimization problem. The Bayesian optimizer builds an initial Gaussian Process (GP) model as a global surrogate for the response function using Latin Hypercube Sampling. Then, the algorithm intelligently selects additional samples to be added to the samples set. The updated samples set is used to create a new GP model in subsequent iterations. Each new sample (set of shape parameters) requires the solution to a high-fidelity shape optimization problem[1]. The samples are selected based on how much they are expected to improve the current best solution to the shape optimization problem. When this expected improvement is acceptably small, the optimal solution has been found.

4.3.1 Input Deck

The surrogate-based shape optimization approach requires two input decks, one input deck for the surrogate-based optimizer and the second input deck for the finite element simulation.The next section will discuss the input deck for the surrogate-based optimizer.

[1] High-fidelity is defined in this manuscript as any operation that requires the solution of a computationally intensive process. In the context of shape optimization, a high-fidelity shape optimization problem uses finite element simulations to compute the quantities of interests needed to evaluate the design criteria.

Surrogate-based optimization

The input deck setup for the surrogate-based shape optimization study is presented in Listing 4.3.1. The concurrent_evaluations keyword is used to set the number of concurrent response evaluations. Meaning, how many finite element simulations are launched concurrently to harvest and add a new set of high-fidelity design criteria evaluations to the samples set. The copy_files keyword is used to copy mandatory files into the run.{itr} directories. The run.{itr} directories are located inside the work directory (workdir). The work directory is created by the optimizer inside the run directory, which is the directory from where the shape optimization study is launched. The {itr} keyword is replaced with the numerical value associated with a major optimization iteration.

Additional Scripts

Two additional scripts are needed to run the surrogate-based shape optimization study. The first script is the *analysis driver script* and the second script is the *postprocessing script*.

Analysis driver

The analysis_driver.sh script contains the sequence of instructions for:

- performing the parameter substitutions,
- updating the computational geometry and the finite element mesh,
- running the linear mechanical analysis, and
- writing the text file containing the numerical values for the design criteria.

The following text snippet shows the setup used in *Example 2* for the analysis_driver.sh script:

Listing 4.3.2: Analysis driver script for the surrogate-based shape optimization problem.

```
#!/bin/bash
dprepro $1 esp_geometry_file.temp esp_geometry_file.csm
plato-cli geometry esp --input esp_geometry_file.csm --output-model esp_geometry_file_
↪opt.csm --output-mesh esp_geometry_file.exo --tesselation esp_geometry_file.eto
python runmorphorm.py mech_analysis_ex2.i
python postprocess.py
mv results.out $2
```

The first line instructs the operating system to use bash as a command interpreter. The second line calls the application performing the parameter substitutions and creating the input deck for the ESP application. The third line calls the ESP application to update the geometry and generate the finite element mesh for the linear mechanical analysis. The fourth line calls the simulation toolkit to simulate the linear mechanical physics and evaluates the design criteria. The fith line *postprocess* the output from the simulation and writes the results.out file containing the numerical values for the design criteria. The last line renames the results.out file to the name recognized by the surrogate-based optimizer.

Postprocess results

The postprocess.py script writes the results.out text file containing the numerical values for the design criteria. The surrogate-based optimizer will update the shape parameters based on the numerical values parsed from the results.out text file. The following text snippet is a typical setup for the postprocess.py script:

Listing 4.3.3: Postprocess script for the surrogate-based shape optimization problem.

```
import itertools
# set filenames
#
tMyFileList=["opt_criteria_history.csv"]
# open output file
#
tOutFile = open("results.out", "w+")
# loop over files and read its data
#
for tMyFile in tMyFileList:
  # open file
  #
  with open(tMyFile) as tFile:
    # read lines
    #
    tLines = tFile.readlines()
    # loop over lines
    #
    for tLine in tLines:
      # split line
      #
      tMyList = tLine.split(', ')
      # write last criterion value to output file
      #
     if(tMyList[0] == "My Objective"):
       tOutFile.write(tMyList[len(tMyList)-2])
   # close read file
   #
   tFile.close()
# close output file
#
tOutFile.close();
```

The `opt_criteria_history.csv` file is written by the the simulation toolkit, i.e., `Analyze`. It records the objective and constraint(s) function evaluation histories.

Analysis

The input deck setup used by the surrogate-based optimizer for the linear mechanical simulation is as follows:

Listing 4.3.4: Input deck setup used by the surrogate-based optimizer for the linear mechanical simulation.

```
begin service 1
  code analyze
  number_processors 1
end service

begin criterion 1
  type mechanical_compliance
end criterion
```

(continues on next page)

(continued from previous page)

```
begin criterion 2
  type volume
  minimum_ersatz_material_value 0
end criterion

begin scenario 1
  physics steady_state_mechanics
  service 2
  dimensions 3
  loads 1 2
  boundary_conditions 1
  material 1
  linear_solver 1
end scenario

begin linear_solver 1
  solver_package tacho
end linear_solver

begin objective
  type weighted_sum
  criteria 1 2
  services 2 2
  scenarios 1 1
  weights 1 0.8
end objective

begin output
  service 2
  output_frequency 5
  native_service_output true
  data dispx dispy dispz
end output

begin boundary_condition 1
  type fixed_value
  location_type sideset
  location_name center_axis
  degree_of_freedom dispx dispz dispy
  value 0 0 0
end boundary_condition

begin load 1
  type traction
  location_type sideset
  location_name left_axis
  value 0 450 0
end load

begin load 2
  type traction
```

(continues on next page)

(continued from previous page)

```
  location_type sideset
  location_name right_axis
  value 0 225 0
end load

begin block 1
  material 1
end block

begin material 1
  material_model isotropic_linear_elastic
  poissons_ratio 0.33
  youngs_modulus 29007548
end material

begin study
  method analysis
  evaluate_criteria true
end study

begin mesh
  name shape_optimization_mech_ex2.exo
end mesh
```

Geometry

The input deck setup for the Engineering Sketch Pad application [Haimes and Dannenhoffer], e.g., the `esp_geometry_file.temp` file presented in the *Analysis driver* section - Listing 4.3.2, is presented next for completeness:

Listing 4.3.5: Input deck setup for the Engineering Sketch Pad application.

```
# v2.csm written by ocsmSave (v1.16)
#
# Constant, Design, and Output Parameters:
despmtr Py {p1} lbound 2.0 ubound 3.0 initial {p1}
despmtr Boffset {p2} lbound 0.05 ubound 0.38 initial {p2}
despmtr Lx {p3} lbound 5.0 ubound 8.0 initial {p3}
despmtr Rx {p4} lbound 5.0 ubound 8.0 initial {p4}
despmtr Px {p5} lbound 1.6 ubound 2.4 initial {p5}

conpmtr Pz 2.0 # lbound 1.6 ubound 2.4
conpmtr Rcap 1 # lbound 1.0 ubound 1.0
conpmtr Rp 0.7 # lbound 0.7 ubound 0.7
conpmtr Rpo 0.5 # lbound 0.5 ubound 0.5
conpmtr Rlo 0.3 # lbound 0.3 ubound 0.3
conpmtr Po 0.2 # lbound 0.2 ubound 0.2

# Global Attributes:
set MeshLength Px/4
```

(continues on next page)

(continued from previous page)

```
# Branches:
skbeg     0   0   0   1
   skvar     xy   0.000000;0.000000;0.000000;2.000000;0.000000;0.000000;8.000000;1.
↪000000;0.000000;2.000000;2.000000;0.000000;0.000000;2.000000;0.000000;-3.000000;1.
↪000000;0.000000;
   skcon     X   1   -1  0
   skcon     Y   1   -1  0
   skcon     H   1   2  0
   skcon     H   4   5  0
   skcon     L   1   2  Px
   skcon     L   4   5  Px
   skcon     Y   5   -1  Py
   skcon     Y   3   -1  Py/2.0
   skcon     Y   6   -1  Py/2.0
   skcon     X   3   -1  Px+Rx
   skcon     X   6   -1  -Lx
   skcon     X   5   -1  0
   linseg    ::x[2]   ::y[2]   0
   linseg    ::x[3]   ::y[3]   0
   linseg    ::x[4]   ::y[4]   0
   linseg    ::x[5]   ::y[5]   0
   linseg    ::x[6]   ::y[6]   0
   linseg    ::x[1]   ::y[1]   0
skend     0
extrude   0   0   Pz
fillet    Py/4.0   7;8;4;5   0
fillet    Rcap   3;8;3;4;6;7;5;6   0
box       Boffset   0   Boffset   Px-2*Boffset   Py   Pz-2*Boffset
fillet    Boffset   1;5;2;5;1;6;2;6   0
subtract  none   1   0
cylinder  Px/2.0   Py/2.0   -Po   Px/2.0   Py/2.0   Pz+Po   Rp
union     0   0   0
cylinder  Px/2.0   Py/2.0   -Po   Px/2.0   Py/2.0   Pz+Po   Rpo
subtract  none   1   0
cylinder  Rx/2.0+Px   Py/2.0   0   Rx/2.0+Px   Py/2.0   Pz   Rlo
subtract  none   1   0
box       Px+Boffset   0.0   Boffset   3.0*Rx/8.0   Py   Pz-2*Boffset
fillet    Boffset   1;5;2;5;1;6;2;6   0
subtract  none   1   0
box       -3.0*Lx/8.0   0.0   Boffset   3.0*Lx/8.0-Boffset   Py   Pz-2*Boffset
fillet    Boffset   1;5;2;5;1;6;2;6   0
subtract  none   1   0
cylinder  -Lx/2.0   Py/2.0   0.0   -Lx/2.0   Py/2.0   Pz   Rlo
subtract  none   1   0

select    body   28
select    face   27   3   1
attribute capsGroup   $left_axis

select    body   28
select    face   27   4   1
attribute capsGroup   $left_axis
```

(continues on next page)

(continued from previous page)

```
select    body   28
select    face   19   3   1
attribute capsGroup   $right_axis

select    body   28
select    face   19   4   1
attribute capsGroup   $right_axis

select    body   28
select    face   17   4   1
attribute capsGroup   $center_axis

select    body   28
select    face   17   3   1
attribute capsGroup   $center_axis

end
```

The *shape parameter descriptors* are shown in Listing 4.3.5 within brackets `{ }`. The `{ }` delimiters are used by the inline expression parser to identify which parameters must be updated with numerical values at runtime. The input deck for the ESP application must be constructed within the ESP environment. The Morphorm software is capable of generating the `exodus mesh` needed for the simulation from the instructions in Listing 4.3.5.

4.3.2 Run Study

To run the shape optimization study from a terminal window, type the following command on the terminal window:

```
python runmorphorm.py shape_optimization_mech_ex2.i
```

`runmorphorm.py` is the python script holding the sequence of instructions to run the Morphorm software and `shape_optimization_mech_ex1.i` is the *input deck* for the surrogate-based shape optimization study. The shape optimization study should run after pressing the `Enter` key.

Table 4.3.1: Shape parameters for the optimized configuration.

Descriptor	Value (in)
p1	2.217958481
p2	0.190976242
p3	6.064035679
p4	6.085564356
p5	1.917134368

4.3.3 Results

Fig. 4.3.1 shows the optimized configuration overlaid with the baseline configuration. The volume for the optimized and baseline configurations are 12.85 in^3 and 13.75 in^3, respectively. The volume of the optimize configuration is 6.5 percent less than the volume of the baseline configuration. The peak displacement for the optimized and baseline configurations were measured at 0.0058 in and 0.0066 in, respectively. As for the von Mises stress, the peak von Mises stress for the optimized and baseline configurations were measured at 17280 psi and 17348 psi. Therefore, the optimizer achieve the design goals while limiting the number of high-fidelity evaluations to sixty. Fig. 4.3.2 shows the displacement and von Mises stress fields for the optimized configuration. Table 4.3.1 shows the shape parameter values for the optimized configuration.

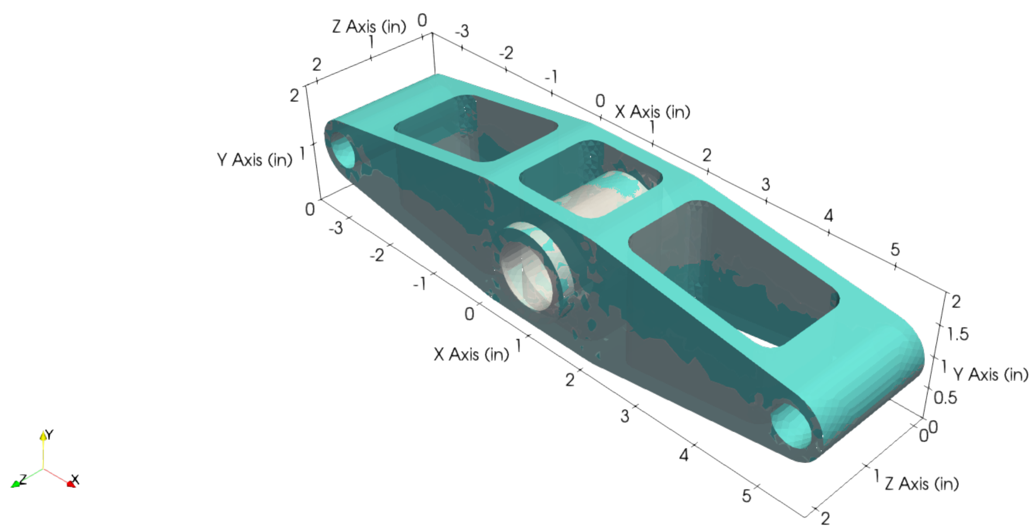

Fig. 4.3.1: The optimized configuration (cyan) overlaid the baseline configuration (gray).

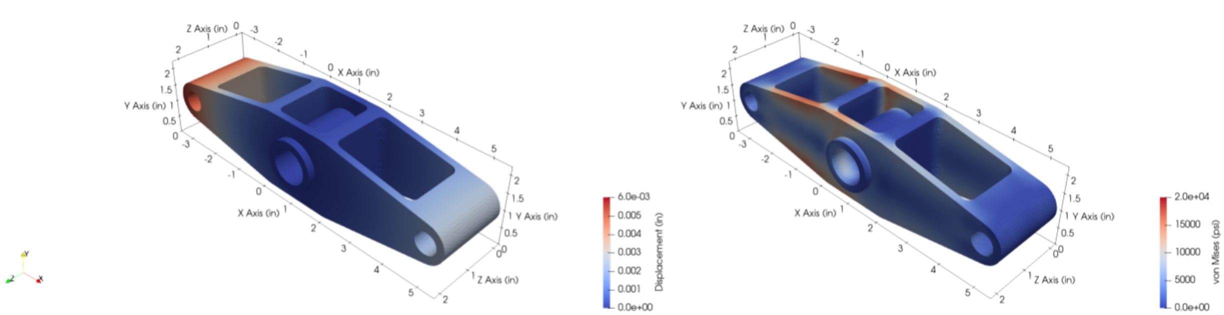

Fig. 4.3.2: The displacement (left pane) and von Mises stress (right pane) fields for the optimized configuration.

BIBLIOGRAPHY

[BS13] Martin Philip Bendsoe and Ole Sigmund. *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2013.

[Bou01] Blaise Bourdin. Filters in topology optimization. *International journal for numerical methods in engineering*, 50(9):2143–2158, 2001.

[GLAP21] Oliver Giraldo-Londono, Miguel A Aguilo, and Glaucio H Paulino. Local stress constraints in topology optimization of structures subjected to arbitrary dynamic loads: a stress aggregation-free approach. *Structural and Multidisciplinary Optimization*, 64:3287–3309, 2021.

[HD13] Robert Haimes and John Dannenhoffer. *The Engineering Sketch Pad: A Solid-Modeling, Feature-Based, Web-Enabled System for Building Parametric Geometry*. American Institute of Aeronautics and Astronautics, 2013. URL: https://arc.aiaa.org/doi/abs/10.2514/6.2013-3073, arXiv:https://arc.aiaa.org/doi/pdf/10.2514/6.2013-3073, doi:10.2514/6.2013-3073.

[JGMK21] S Bestley Joe, T Gomathi, D Marshiana, and N R Krishnamoorthy. Effects of joule heating on the temperature-voltage relationship in copper. In *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, volume, 400–403. 2021. doi:10.1109/ICICT50816.2021.9358523.

[Sja92] G D Sjaardema. Aprepro: an algebraic preprocessor for parameterizing finite element analyses. 12 1992. URL: https://www.osti.gov/biblio/6816618.

[SU11] Gregory Sjaardema and USDOE. Aprepro - algebraic preprocessor, version 2.0. 2 2011. URL: https://www.osti.gov/biblio/2403459, doi:10.11578/dc.20171025.1039.

[Sja17] Gregory D. Sjaardema. Sandia engineering analysis code access system v. 2.0.1, version 00. 10 2017. URL: https://www.osti.gov//servlets/purl/1398883.

[SSD+22] Clint Stimpson, Gregory Sjaardema, Clifton Dudley, Robert Yorgason, Byron Hanks, David White, Robert Kerr, Scott Mitchell, Jason Shepherd, Timothy Tautges, Darryl Melander, Trevor Hensley, Paul Stallings, Teddy Blacker, Michael Borden, Brett Clark, Leslie Fortier, Corey McBride, Philippe Pebay, Matthew Staten, Craig Vineyard, Kevin Pendley, Steven Owen, Corey Ernst, William Quadros, Neal Grieb, Michael Plooster, Mark Richardson, Steve Storm, John Fowler, Mike Stephenson, Rammagy Yoeu, Karl Merkley, Ray Meyers, Mark Dewey, Sara Richards, Randy Morris, Ved Vyas, Boyd Tidwell, Erick Johnson, Mike Parrish, Adam Woodbury, John Malone, Jane Hu, Ben Aga, Andrew Rout, and USDOE. Cubit v.16.x, version 16.x. 8 2022. URL: https://www.osti.gov/biblio/1894127, doi:10.11578/dc.20221017.1.

[Sva87] Krister Svanberg. The method of moving asymptotes—a new method for structural optimization. *International journal for numerical methods in engineering*, 24(2):359–373, 1987.