Patrick Jaap

On plastic deformation: From physics over convex analysis to numerical simulation

Lecture 3 // Dresden, July 2, 2019

Goals of this Graduate Lecture

Why are we (still) here and what will we (also) learn?

Goals of this Graduate Lecture

Why are we (still) here and what will we (also) learn?

1st lecture (two weeks ago)

- developed (an easy) physical model of elasticity and plasticity
- considered engineering approaches for typical materials (e.g. steel)

Goals of this Graduate Lecture

Why are we (still) here and what will we (also) learn?

1st lecture (two weeks ago)

- developed (an easy) physical model of elasticity and plasticity
- considered engineering approaches for typical materials (e.g. steel)

2nd lecture (last week)

- proofed some nice results from convex analysis

Goals of this Graduate Lecture

Why are we (still) here and what will we (also) learn?

1st lecture (two weeks ago)

- developed (an easy) physical model of elasticity and plasticity
- considered engineering approaches for typical materials (e.g. steel)

2nd lecture (last week)

- proofed some nice results from convex analysis

3rd lecture (today)

- formulate two coupled systems (dual, primal)
- give hints for implementations

Summary – basic equations

- Stress, Displacement

$$\sigma \in \mathbb{R}^{3\times 3}_{sym}, \quad u \in \mathbb{R}^3$$

- Strain

$$\epsilon = e + p = \frac{1}{2}\left(\nabla u + \nabla u^T\right) \in \mathbb{R}^{3\times 3}_{sym}$$

- Yield function with hardening

$$\Phi(\sigma, \alpha, g) = \phi(\sigma + \alpha) + g \leq 0$$

- Law of equilibrium:

$$-\operatorname{div} \sigma = f$$

- Hooke's law:

$$\exists \mathbf{C} \in \mathbb{R}^{(3\times 3)\times(3\times 3)} : \; \sigma = \mathbf{C} e$$

- Maximum work

$$\dot{p} \in N_E(\sigma)$$

Summary – generalized variables

- Generalized Stress

$$\Sigma := (\sigma, \alpha, g) \in \mathbb{R}^{3\times 3}_{sym} \times \mathbb{R}^{3\times 3}_{sym} \times \mathbb{R}$$

- Generalized Strain

$$P := (p, a, \eta) \in \mathbb{R}^{3\times 3}_{sym,0} \times \mathbb{R}^{3\times 3}_{sym,0} \times \mathbb{R}_{\geq 0}$$

- Yield function with hardening

$$\Phi(\sigma, \alpha, g) = \phi(\sigma + \alpha) + g$$

defining the elastic region

$$\mathcal{E} = \{\Sigma : \ \Phi(\Sigma) \leq 0\}$$

- we concluded

$$a = p$$

- typical relations

$$\alpha = -k_1 p$$
$$g = -k_2 \eta$$

- We proofed in just 50 minutes

$$\dot{P} \in N_{\mathcal{E}}(\Sigma) \quad \Leftrightarrow \quad \Sigma \in \partial D(\dot{P})$$

- How do we couple these equations?

1st try: dual formulation

let us begin with the first ansatz: *dual formulation*

- here, we consider stress σ and displacement u as unknowns
- starting point is the line

$$\dot{P} \in N_{\mathcal{E}}(\Sigma)$$

- this is by definition

$$\langle \dot{P}, \tilde{\Sigma} - \Sigma \rangle \leq 0 \qquad \forall \tilde{\Sigma} \in \mathcal{E}$$

1st try: dual formulation

let us begin with the first ansatz: *dual formulation*

- here, we consider stress σ and displacement u as unknowns
- starting point is the line

$$\dot{P} \in N_{\mathcal{E}}(\Sigma)$$

- this is by definition

$$\langle \dot{P}, \tilde{\Sigma} - \Sigma \rangle \leq 0 \qquad \forall \tilde{\Sigma} \in \mathcal{E}$$

- but first, we define some (bi-)linear functions and stick them together

dual formulation – (bi-)linear functions

- for the force field f we had

$$-\operatorname{div}\sigma = f$$

- with that in mind, we define a linear functional $l(t)$

$$\langle l(t), u\rangle := -\int_\Omega f(t)\cdot u\,dx$$

- and a bilinear form $b(\cdot,\cdot)$

$$b(u,\sigma) := -\int_\Omega \epsilon(u) : \sigma\,dx$$

- this yields with integration by parts

$$b(\tilde{u},\sigma) = \langle l(t), \tilde{u}\rangle \qquad \forall \tilde{u}$$

dual formulation – more bilinear functions

from now on, we assume the Hooke tensor **C** to be invertible and symmetric

- stress – stress mapping:

$$a(\sigma, \tilde{\sigma}) := \int_\Omega \sigma : \mathbf{C}^{-1} \tilde{\sigma} \, dx$$

- and for the generalized variables

$$c_1(\alpha, \tilde{\alpha}) := \int_\Omega \frac{1}{k_1} \alpha : \tilde{\alpha} \, dx$$
$$c_2(g, \tilde{g}) := \int_\Omega \frac{1}{k_2} g \cdot \tilde{g} \, dx$$

- resulting in

$$A(\Sigma, \tilde{\Sigma}) := a(\sigma, \tilde{\sigma}) + c_1(\alpha, \tilde{\alpha}) + c_2(g, \tilde{g})$$

dual formulation

- back to

$$\int_\Omega \langle \dot P, \tilde\Sigma - \Sigma\rangle \, dx \le 0 \qquad \forall \tilde\Sigma \in \mathcal{E}$$

- this is

$$\int_\Omega \dot p : (\tilde\sigma - \sigma) + \dot a : (\tilde\alpha - \alpha) + \dot\eta \cdot (\tilde g - g) \, dx \le 0$$

- and now we use

$$\dot\alpha = -k_1 \dot a, \quad \dot g = -k_2 \dot\eta$$

- and for $\dot p$ with Hooke's law

$$\epsilon(\dot u) = \dot p + \dot e = \dot p + \mathbf{C}^{-1}\dot\sigma \quad \Rightarrow \quad \dot p = \epsilon(\dot u) - \mathbf{C}^{-1}\dot\sigma$$

dual formulation

- this is

$$\int_\Omega \dot{p} : (\tilde{\sigma} - \sigma) + \dot{a} : (\tilde{\alpha} - \alpha) + \dot{\eta} \cdot (\tilde{g} - g)\, dx \leq 0$$

- and now we use

$$\dot{\alpha} = -k_1 \dot{a}, \quad \dot{g} = -k_2 \dot{\eta}, \quad \dot{p} = \epsilon(\dot{u}) - \mathbf{C}^{-1}\dot{\sigma}$$

- which means

$$\int_\Omega \underbrace{\epsilon(\dot{u}) : (\tilde{\sigma} - \sigma)}_{-b(\dot{u},\tilde{\sigma}-\sigma)} - \underbrace{\mathbf{C}^{-1}\dot{\sigma} : (\tilde{\sigma} - \sigma)}_{a(\dot{\sigma},\tilde{\sigma}-\sigma)} - \underbrace{\frac{1}{k_1}\dot{\alpha} : (\tilde{\alpha} - \alpha)}_{c_1(\dot{\alpha},\tilde{\alpha}-\alpha)} - \underbrace{\frac{1}{k_2}\dot{g} \cdot (\tilde{g} - g)}_{c_2(\dot{g},\tilde{g}-g)}\, dx \leq 0$$

- therefore,

$$A(\dot{\Sigma}, \tilde{\Sigma} - \Sigma) + b(\dot{u}, \tilde{\sigma} - \sigma) \geq 0 \quad \forall \tilde{\Sigma} \in \mathcal{E}$$

dual formulation – time discrete

- if we approximate the time derivative by finite (implicit) differences

$$A(\dot{\Sigma}, \tilde{\Sigma} - \Sigma) + b(\dot{u}, \tilde{\sigma} - \sigma) \geq 0 \quad \forall \tilde{\Sigma} \in \mathcal{E}$$

- we will get

$$A(\Sigma_n - \Sigma_{n-1}, \tilde{\Sigma} - \Sigma_n) + b(u_n - u_{n-1}, \tilde{\sigma} - \sigma_n) \geq 0 \quad \forall \tilde{\Sigma} \in \mathcal{E}$$

- and don't forget

$$b(\tilde{u}, \sigma_n) = \langle l_n, \tilde{u} \rangle \qquad \forall \tilde{u}$$

dual formulation – time discrete

- if we approximate the time derivative by finite (implicit) differences

$$A(\dot{\Sigma}, \tilde{\Sigma} - \Sigma) + b(\dot{u}, \tilde{\sigma} - \sigma) \geq 0 \quad \forall \tilde{\Sigma} \in \mathcal{E}$$

- we will get

$$A(\Sigma_n - \Sigma_{n-1}, \tilde{\Sigma} - \Sigma_n) + b(u_n - u_{n-1}, \tilde{\sigma} - \sigma_n) \geq 0 \quad \forall \tilde{\Sigma} \in \mathcal{E}$$

- and don't forget

$$b(\tilde{u}, \sigma_n) = \langle l_n, \tilde{u} \rangle \qquad \forall \tilde{u}$$

- how do we solve this problem?

dual formulation – solution algorithms

- how do we solve this problem?
- there are countless numerical schemes in literature...
- let’s look at the ”classical” strategy, which is the base of most schemes

dual formulation – solution algorithms

- how do we solve this problem?
- there are countless numerical schemes in literature...
- let's look at the "classical" strategy, which is the base of most schemes

- in each time step, we have to find Σ_n and u_n

$$\begin{aligned} A(\Sigma_n - \Sigma_{n-1}, \tilde{\Sigma} - \Sigma_n) + b(u_n - u_{n-1}, \tilde{\sigma} - \sigma_n) &\geq 0 && \forall \tilde{\Sigma} \in \mathcal{E} \\ b(\tilde{u}, \sigma_n) &= \langle l_n, \tilde{u} \rangle && \forall \tilde{u} \end{aligned}$$

- we will *predict* an (incorrect) displacement and *correct* with a suitable stress until convergence
- $\Rightarrow$ this is called *predictor–corrector* method

Predictor

let u_n^i and Σ_n^i denote the currently known iterates

- we assume that we have an elastic process (may be wrong, of course):

$$\begin{aligned} \sigma_n^{i+1} = \mathbf{C}\epsilon(u_n^{i+1}) &= \mathbf{C}\epsilon(u_n^i + u_n^{i+1} - u_n^i) \\ &= \sigma_n^i + \mathbf{C}\epsilon(u_n^{i+1} - u_n^i) \end{aligned}$$

- this implies

$$\langle l_n, \tilde{u} \rangle = b(\tilde{u}, \sigma_n^{i+1}) = b(\tilde{u}, \sigma_n^i) + \int_\Omega \mathbf{C}\epsilon(u_n^{i+1} - u_n^i) : \epsilon(\tilde{u})\, dx$$

- this is a linear equation in u_n^{i+1}, which can be solved with common methods (no details here, sorry)

Corrector

Of course, assuming an elastic process may have been wrong!

- now we have u_n^{i+1} given and try to find a Σ_n^{i+1}
- we set the so-called *trial stress*

$$\Sigma^{trial} := \Sigma_{n-1} + \left[\mathbf{C}\epsilon(u_n^{i+1} - u_{n-1}), 0, 0\right]^T$$

- and if we plug this in

$$A(\Sigma^{trail} - \Sigma_n^{i+1}, \tilde{\Sigma} - \Sigma_n^{i+1}) \leq 0$$

one can show (due to lack of time, this person is not me) that this forms a suitable pair

$$(\Sigma_n^{i+1}, u_n^{i+1})$$

if the inequality is solved for Σ_n^{i+1}

Corrector

- it was a quadratic optimization, so again, there are efficient methods for solving
- but σ_n^{i+1} may (again) violate the first equation

$$(\tilde{u}, \sigma_n^{i+1}) = \langle l_n, \tilde{u} \rangle \qquad \forall \tilde{u}$$

- so we keep repeating the predictor–corrector algorithm until both (in-)equalities are fulfilled
- under some elliptic assumptions on **C**, the algorithm will converge

Corrector

- it was a quadratic optimization, so again, there are efficient methods for solving
- but σ_n^{i+1} may (again) violate the first equation

$$(\tilde{u}, \sigma_n^{i+1}) = \langle l_n, \tilde{u} \rangle \qquad \forall \tilde{u}$$

- so we keep repeating the predictor–corrector algorithm until both (in-)equalities are fulfilled
- under some elliptic assumptions on **C**, the algorithm will converge
- problem: the algorithm is slow in practice

2nd try: primal formulation

- in the second lecture, we proofed

$$\dot{P} \in N_{\mathcal{E}}(\Sigma) \quad \Leftrightarrow \quad \Sigma \in \partial D(\dot{P})$$

- D was the dissipation function of the yield function:

$$D(\dot{P}) := \sup_{\Sigma \in \mathcal{E}} \{\dot{P} : \Sigma\}$$

- D can be expressed directly for Von Mises

$$D(\dot{P}) = \begin{cases} \sigma_0 \|\dot{p}\|_F & \|\dot{p}\|_F \leq \eta \\ \infty & \text{else} \end{cases}$$

- and for Tresca flow rule

$$D(\dot{P}) = \begin{cases} \sigma_0 \|\dot{p}\|_2 & \|\dot{p}\|_2 \leq \eta \\ \infty & \text{else} \end{cases}$$

2nd try: primal formulation

- the formulation

$$\Sigma \in \partial D(\dot{P})$$

  is defined by

$$D(\tilde{P}) \geq D(\dot{P}) + \Sigma : (\tilde{P} - \dot{P}) \qquad \forall \tilde{P}$$

- which is

$$D(\tilde{P}) \geq D(\dot{P}) + \sigma : (\tilde{p} - \dot{p}) + \alpha(\tilde{a} - \dot{a}) + g \cdot (\tilde{\eta} - \dot{\eta}) \qquad \forall \tilde{P}$$

- again, we use $\alpha = -k_1 a$, $g = -k_1 \eta$ and $a = p$!
- for σ, we use

$$\sigma = \mathbf{C}e = \mathbf{C}(\epsilon(u) - p)$$

2nd try: primal formulation

- we had

$$D(\tilde P) \geq D(\dot P) + \sigma : (\tilde p - \dot p) + \alpha(\tilde a - \dot a) + g \cdot (\tilde \eta - \dot \eta) \qquad \forall \tilde P$$

- with $\alpha = -k_1 a$, $g = -k_1 \eta$ and $a = p$!
- fand

$$\sigma = \mathbf{C} e = \mathbf{C}(\epsilon(u) - p)$$

- and therefore

$$\int_\Omega D(\tilde P)\, dx \geq \int_\Omega D(\dot P) + \mathbf{C}(\epsilon(u) - p) : (\tilde p - \dot p) - k_1 p : (\tilde p - \dot p) - k_2 \eta \cdot (\tilde \eta - \dot \eta)\, dx \qquad \forall \tilde P$$

2nd try: primal formulation

- and therefore

$$\int_\Omega D(\tilde{P})\,dx \geq \int_\Omega D(\dot{P}) + \mathbf{C}(\epsilon(u) - p) : (\tilde{p} - \dot{p}) - k_1 p : (\tilde{p} - \dot{p}) - k_2 \eta \cdot (\tilde{\eta} - \dot{\eta})\,dx \qquad \forall \tilde{P}$$

- now, use also that

$$-\operatorname{div} \sigma \cdot (\tilde{u} - \dot{u}) = f \cdot (\tilde{u} - \dot{u})$$

- integrated by parts

$$\int_\Omega \mathbf{C}(\epsilon(u) - p) : (\epsilon(\tilde{u}) - \epsilon(\dot{u}))\,dx = \int_\Omega \sigma : (\epsilon(\tilde{u}) - \epsilon(\dot{u}))\,dx = \int_\Omega f \cdot (\tilde{u} - \dot{u})\,dx$$

primal formulation

- subtracting the last two (in-)equalities leads to

$$-\int_\Omega f\cdot(\tilde{u}-\dot{u})\,dx \geq \int_\Omega D(\dot{P})-D(\tilde{P})-\mathbf{C}(\epsilon(u)-p):((\epsilon(\tilde{u})-\epsilon(\dot{u}))-(\tilde{p}-\dot{p}))$$
$$-k_1 p:(\tilde{p}-\dot{p})-k_2\eta\cdot(\tilde{\eta}-\dot{\eta})\,dx \qquad \forall\tilde{P},\tilde{u}$$

- then, we set $w=(u,p,\eta)$ and define new (!) (bi-)linear function that represent

$$-\langle l(t),\tilde{w}-\dot{w}\rangle \geq j(\dot{w})-j(\tilde{w})-a(w,\tilde{w}-\dot{w}) \quad \forall\tilde{w}$$

- or rearranged

$$a(w,\tilde{w}-\dot{w})+j(\tilde{w})-j(\dot{w}) \geq \langle l(t),\tilde{w}-\dot{w}\rangle \quad \forall\tilde{w}$$

primal formulation – time discrete

- like before, we replace the time derivative by an implicit finite difference

$$a(\Delta w_n, \tilde{w} - \Delta w_n) + j(\tilde{w}) - j(\Delta w_n) \geq \underbrace{\langle l_n, \tilde{w} - \Delta w_n \rangle}_{\text{contains part of } a} \quad \forall \tilde{w} \qquad (1)$$

  with

$$\Delta w_n = w_n - w_{n-1}$$

- now we can use a result from convex optimization

primal formulation – time discrete

- like before, we replace the time derivative by an implicit finite difference

$$a(\Delta w_n, \tilde{w} - \Delta w_n) + j(\tilde{w}) - j(\Delta w_n) \geq \underbrace{\langle l_n, \tilde{w} - \Delta w_n \rangle}_{\text{contains part of } a} \quad \forall \tilde{w} \qquad (1)$$

  with

$$\Delta w_n = w_n - w_{n-1}$$

- now we can use a result from convex optimization

Theorem

Solving the inequality (1) for Δw_n is equivalent to minimizing

$$L(\Delta w_n) := \frac{1}{2} a(\Delta w_n, \Delta w_n) + j(\Delta w_n) - \langle l_n, \Delta w_n \rangle$$

primal formulation – summary

Theorem

Solving the inequality (1) for Δw_n is equivalent to minimizing

$$L(\Delta w_n) := \frac{1}{2}a(\Delta w_n, \Delta w_n) + j(\Delta w_n) - \langle l_n, \Delta w_n \rangle$$

- to compute the increment, we have to minimize a strictly convex function
- we get uniqueness and existence of solution for free
- but: L is not smooth everywhere!
- so classical minimizing algorithms won't work, we need special solvers

Summary

- we have seen two different formulations:
- primal:
  - starting from $\Sigma \in \partial D(\dot{P})$
  - resulting in an convex minimization problem
  - efficient solvers are available (ask me ☺)
- dual:
  - starting from $\dot{P} \in N_{\mathcal{E}}(\Sigma)$
  - leading to optimization inequalities
  - algorithms are mostly of predictor–corrector type
  - but easy to implement

Summary – overall

We did it!

- we have seen the engineering approach of plasticity
- we considered different elastic regions
- we have seen hardening rules
- an important theorem from convex analysis was proofed
- different approaches of solving the complete system were given
- Slides can be found here:

  `https://www.math.tu-dresden.de/~jaap/`

- contact me for the simulations, programs, etc.

Sources

- plasticity theory:
  - Weimin Han & B. Daya Reddy, Plasticity, Mathematical Theory and Numerical Analysis, Second Edition, Springer Science+Business Media, LLC 2013
- convex analysis/optimization
  - Ivar Ekeland & Roger Temam, Convex Analysis and Variational Problems, North-Holland Publishing Company, 1973