WEEK 3 · SELF-STUDY PACK

Three hours, on your own

Networks, flows and process superstructures

2105623 Optimization of Chemical Processes · Semester 1 / 2026 · Monday 24 August 2026

I am away on Monday 24 August. **There is no lecture and nobody in the room**, so this pack is the session: the same three hours, the same three activities, done by you instead of with me. Everything you need is on the course page:

www.skhgroup.net/teaching/2105623/

w03-slides.pdf — the slides, 32 of them
w03-transportation.xlsx — two sheets, both used today
w03-modeling2-networks.ipynb — the notebook
w02-feedback.pdf — the Week 2 answers. Read this one first.

The one rule that makes this pack work. The slide deck contains the answers to all three activities, on slides **13**, **21** and **29**. With no instructor to hold them back, you are holding them back yourself. **Do the activity first, on paper, and only then turn to the answer slide.** An answer you read before you tried is worth nothing — it feels like understanding and it is not. Where this pack says **stop**, stop.

Two things about the slides. They were written for the classroom version of this session, so wherever a slide gives *logistics* — work in pairs, hand back one sheet, run every section of the notebook, install only `ipopt` — **this pack overrides it**. And the slides call the files by their archive names. Same files, two names:

the slides say	you have
`W03_transportation_STUDENT.xlsx`	`w03-transportation.xlsx`
`W03_modeling2_networks.ipynb`	`w03-modeling2-networks.ipynb`
`W03_modeling2_networks.pdf`	`w03-slides.pdf`

Do this in the first two minutes, before Part A. Start the two installs in Part F *now* and leave them running while you do the paper parts — a `conda` install can take twenty minutes on its own, and Part A's optional task and all of Part F need them. Come back to Part F at the end to record the result.

And if you run out of time, drop work in this order: keep **Part F** (Week 4 cannot run without it) and the three predictions **B3**, **B5** and **C3**; then C5; then D; then the optional task in A. Say in what you hand in what you dropped — that is not a penalty, it is information I need.

How to run the three hours. The room is free if you want it. Parts A and E are better in pairs or threes — they are drawing tasks and the argument is the point. Parts B and C are individual: you need the software working on *your* machine before Week 4. The times are a guide, not a budget; three hours of unbroken work is not realistic, so take a break where the classroom version had one, after Part C.

Part A The node balance is the model (35 min)

Read slides **1 to 11**. They set up one template — the minimum cost flow problem — and show that transportation, transshipment, shortest path and maximum flow are four choices of data inside it. Then do the first activity from slide **12**, on paper:

A1 Draw the node-arc diagram of the network described on slide 12 by hand. Label every arc with its cost and its capacity.

A2 Mark each node with its net supply b_n, and check that the eight values sum to zero. Say in one sentence why they must.

A3 Write the node balance for hub $H1$ in full, in the convention *outflow minus inflow equals* b_n. Every arc that touches $H1$ must appear exactly once, on the correct side.

Stop. Only now open slide 13. Compare, and answer in writing: **A4** did your diagram have the same *incidence structure* as the answer, even if it looked completely different on the page? **A5** did you write the reverse arc $H2 \to H1$ as a separate arc with its own non-negative flow, or did you treat it as negative flow on $H1 \to H2$? Say which, and why only one of the two is a model.

Optional, if you are ahead. The slide-12 network is small enough to solve. Put it in the notebook and check that the minimum cost is **19,570.00 USD/week**, and that the inter-terminal arc $H1 \to H2$ carries 40 t at the optimum — the arc most pairs forget to draw is the one that is used.

Part B Transportation in the spreadsheet (40 min)

Open **w03-transportation.xlsx**, sheet **Transportation**, and have OpenSolver loaded. The Readme sheet gives the exact text of every formula you need and the full solver setup, field by field.

B1 Complete the yellow cells — the row sums `G12:G14`, the column sums `C15:F15`, and the objective `C19` as one SUMPRODUCT over the whole matrix. Then solve with CBC:

Objective	`$C$19`, minimise	
Changing cells	`$C$12:$F$14`	
Constraint 1	`$G$12:$G$14 <= $I$12:$I$14`	supply
Constraint 2	`$C$15:$F$15 >= $C$17:$F$17`	demand

Tick Sensitivity Analysis.

The objective cell must read **23,570.00** USD/week. If it does not, you have a formula error, and finding it is the exercise.

B2 Report the shipment plan and the sensitivity report: the supply duals u_i and the demand duals v_j. Every shipment came out a whole number of tonnes from a pure LP with no integer variable anywhere. Say, in one sentence, what property of the constraint matrix caused that.

B3 **Predict before you touch anything.** All three supply rows bind — every plant ships everything it has. For *each* plant in turn, write down, before testing, how much the total cost would fall if that plant could supply one more tonne. Three numbers, on paper. Then test them one at a time and compare, and say what you make of the result.

Slide 9 prints these three duals. If you have already read it — Part A told you to — then say so and write the three numbers down anyway, before you re-solve. The mark is for the numbers being on paper first, not for not having seen them. If you have *not* noticed slide 9, do not go back for it: predict cold, then check, and tell me how you did.

B4 Now break the balance on purpose, as slide 20 asks. Reduce total supply below total demand — set $R2$ to 200 t — and re-solve. Record *exactly* what the solver reports, word for word. Then add a dummy plant with a large penalty cost on every lane, give it the missing **180 t** (1,130 − 950), re-solve, and record the new objective and **which gigafactory the dummy serves**.

Do this on a copy of the sheet, and check one thing. The dummy needs a fourth row in the cost block *and* a fourth row in the flow block, and inserting a row directly below a range does **not** widen it — so `C19` will still read `SUMPRODUCT(C6:F8,C12:F14)` and will silently ignore the dummy. Widen it by hand, and widen the changing cells and both constraint ranges in OpenSolver too. If your answer comes out at exactly 19,610 with no penalty in it, that is the mistake you have made.

B5 The sensitivity report says one more tonne of demand at $G3$ is worth 27 USD. Restore the original data, raise $G3$'s demand from 330 to 331 t, change nothing else, and re-solve. **The model is infeasible.** Explain, in three or four sentences, how both of those statements can be true at once.

Stop. Only now open slide 21, and check B4 against it.

B3 and B5 are the two questions I most want answers to, and they are the same question twice. Both ask what a number from a sensitivity report actually promises you. Write both in your own words, from your own solves, before you look anything up — and do not go back and tidy a prediction after seeing the result. A revised prediction teaches nothing.

Part C Transshipment: capacities, potentials and reduced costs — 40 min

Same workbook, sheet **Transshipment**. Eighteen arcs, nine nodes, a capacity on every arc, and two coastal terminals that are pure transit nodes.

C1 Complete *every* yellow cell — twenty-eight of them. The objective `C25`, which the Readme does not give, so here it is: `=SUMPRODUCT(E6:E23,G6:G23)`. Then the outflow and inflow columns `C29:D37`, using the two SUMIF patterns the Readme *does* give; **and column** `E29:E37`, the difference `=C29-D29` and so on down. Column E is easy to miss and it is the one the solver constraint points at. Solve with `$E$29:$E$37 = $G$29:$G$37` and the arc capacities as upper bounds. Tick Sensitivity Analysis again. **The objective must read 25,850.00 USD/week.**

C2 Which arcs come out **saturated** — flow exactly equal to capacity? Name them from *your* solution. Expect two or three.

Do not expect your flows to match slide 10. This network has *several* different optimal shipping plans, all costing exactly 25,850.00; slide 10 shows one of them and your solver may well find another — one in which the direct lane $R1 \to G1$ carries 50 t, for instance. **The total is the check, not the pattern.** If your objective reads 25,850.00 and every node balance holds, your sheet is right; go on. If the total is anything else, that is when to hunt for the error.

C3 **Predict, in writing, before you test anything.** For *each* saturated arc in your own solution, and in *both* directions: if its capacity were raised by one tonne, how much would the total cost fall — and if it were lowered by one tonne, how much would the cost rise? **Two numbers per arc, on paper**, before you touch the sheet.

C4 Now test them, one at a time, and compare. **Your arcs will not all behave the same way**, and at least one of them will not behave the way a price is supposed to behave at all. Say which, say what it did, and explain it from the network — what is it about that arc's position that makes widening it and narrowing it worth different amounts?

C5 Build the reduced-cost check yourself; the sheet does not have one. Type the nine node potentials π_n from the sensitivity report — the shadow prices of the node balances — into `J29:J37`, beside the node names in `B29:B37`. Then in `H6` enter

```
=E6-(INDEX($J$29:$J$37,MATCH(C6,$B$29:$B$37,0))-INDEX($J$29:$J$37,MATCH(D6,$B$29:$B$37,0)))
```

and fill it down to `H23`. That is $\bar{c}_{ij} = c_{ij} - (\pi_i - \pi_j)$, one row per arc. Now check the rule on all eighteen: a *positive* reduced cost forces the arc to be empty, a *negative* one forces it to be saturated, and only a *zero* reduced cost allows a flow strictly between the bounds. **Report every row where the reduced cost is zero**, and say what a zero on an *empty* arc tells you about the solution you found.

Two warnings about the potentials, and both are the point rather than a nuisance.

One. They are fixed only **up to an added constant**. Add the same number to all nine and every reduced cost is unchanged, because only the *difference* $\pi_i - \pi_j$ ever appears. So when you run the notebook in Part F it will print a different set of nine numbers from your sensitivity report — in my run, every one of them lower by 32 — and **nothing is wrong**. Compare the reduced costs, never the potentials.

Two. On this instance one of the nine is not even fixed up to a constant: it has a whole range of equally valid values. So your list of zero rows may be one row shorter or longer than your neighbour's and neither of you is wrong. Report what *your* solver gave you, and if you can work out which node it is, say so — that is the degeneracy from C4, seen from the dual side.

Do not compare 23,570 with 25,850. They are not two options for the same system — the first has unlimited lane capacity and no terminal handling, the second has a capacity on every arc and forces most flow through a terminal. Slide 11 says why, and the habit it is teaching is: state what a model leaves out before comparing its objective with anything.

Part D Paths, cuts, and integers for free — 20 min

Reading only, slides **14 to 18**. Answer briefly, two or three sentences each:

D1 Shortest path is the same template with only b changed. Why does the LP return a single path rather than material spread thinly over several routes, when nothing in the model says the answer must be a path?

D2 The maximum flow instance delivers 1,100 t against 1,130 t of demand, and the entire 30 t shortfall lands on $G3$. Slide 16 names the five arcs of the minimum cut. Which capacity increases would recover the missing 30 t, and — more importantly — what is the value of increasing the capacity of any arc *not* in the cut?

D3 Slide 18 adds one innocuous-looking row: terminal $H1$ may handle at most 35 percent of everything leaving the plants. Four flows immediately come out at half a tonne. Say what that row did to the constraint matrix, and what it costs you in the class of the model.

Part E The superstructure — 30 min

Read slides **22 to 27**. Then do the third activity, from slide **28**, on one sheet of paper:

E1 Draw the superstructure for the hydrogen problem: every unit, every candidate stream, the mixing and splitting nodes, and the product node.

E2 Mark every decision as *continuous* (how much goes through each route) or *discrete* (whether to build a unit at all), using two clearly different symbols.

E3 State which constraints change if unit selection is added, and write the one new constraint family in algebra.

Stop. Only now open slide 29. Then answer: **E4** the linear model has *no split fraction* in it, and slide 24 says it does not need one. Explain why writing the split as a ratio would be a mistake, and where the split actually comes from. **E5** the LP leaves the electrolyser at zero; the MILP switches it on. In two sentences, why does adding capital charges turn on a route that the cheaper model refused to use at all?

Part F Before Week 4 — 15 min

Week 4 crosses from the spreadsheet to Python and needs a **nonlinear** solver, which nothing you have run so far has required. Install it now, not on the morning of Week 4 — and install a linear one too, because the notebook needs that to run at all:

`conda install -c conda-forge ipopt` or the IDAES extensions installer

`pip install highspy` the linear and mixed-integer solver the notebook looks for first

Then test it and paste the output into what you hand in:

```
from pyomo.environ import SolverFactory
print(SolverFactory('ipopt').available())
```

Open **w03-modeling2-networks.ipynb** and run it **from the top** down to the end of section 7.1 — the first two cells sit above section 1 and define everything else needs, so they cannot be skipped. Confirm that section 2.1 prints the same 23,570.00 as your spreadsheet, to the cent. If the notebook and your sheet ever disagree, the sheet is wrong.

Section 7.2 is optional, and its error message is misleading — read this before you panic. 7.2 solves a nonconvex MINLP. It looks for `bonmin` or `couenne`, and finding neither it falls back to Pyomo's own `MindtPy`, which then drives `ipopt`. So if `ipopt` is not installed, **7.2 stops with** `ValueError: ipopt is not available.` — which reads like a broken installation but is not one. **Your Part F check is the authority.** If `SolverFactory('ipopt').available()` printed `True`, your setup is correct and Week 4 will work, whatever 7.2 does. Skip it, and skip the figure cell right after it as well — that cell uses a variable 7.2 would have created, so it raises `NameError` on its own. Both results are on slides 27 and 29.

What to hand in, and when

One PDF containing **A1–A5**, **B1–B5**, **C1–C5**, **D1–D3**, **E1–E5** and the **F** solver output. Photographs of handwriting and of your two diagrams are fine, and expected. Name the file `2105623-w03-yourID.pdf` and submit it on myCourseVille **before midnight on Friday 28 August**.

Graded as participation, not as a test. The three predictions — B3, B5 and C3 — are worth more to me than any of the totals, and they only work if they are written down before you check. **If you get stuck on the software, do not lose the three hours to it.** Write down the exact error message, do Parts A, D and E on paper, and send me the error by e-mail the same day; I will answer before Week 4 and there will be a short installation clinic at the start of it. **HW1 is due Monday 31 August**, at the start of Week 4, and it covers Weeks 1 to 3. Its Problem 5 asks for the transshipment model of Problem 2 in Pyomo, over an arc set — which is exactly the structure Part C put in front of you, on different data.