Module 5b: Optimizing gas chromatography separations with simplex optimization

Prelab Assignment

1. a) Use Google Scholar to find a paper in the primary scientific literature that reports separating the components of a "BTEX" mixture by gas chromatography. Use the Copley library website (or other site) to access the paper. Upload this paper into your ELN prelab.

   b) List the 6 compounds that are present in a BTEX mixture.

   c) Record the details of the separation described in the paper. What column was used (Length, i.d., stationary phase, and stationary phase thickness)?

   d) What was the mobile phase "carrier" gas and flow rate?

   e) What was the temperature or temperature gradient program?

   f) What solvent focusing or cold trapping used?

2. Set up your ELN Data & Obs section with a goal statement, the procedure summarized in your own words in 2-column format, and some data tables ready to be filled in.

Lab exercise part 1: Simulator accuracy and carrier gas replacement

Have one person in your lab group register on the Restek corporate website to gain access to the EZGC Chromatograph Modeler program (https://www.restek.com/proezgc). Although having to register is lame, this does allow you to save your favorite separation conditions (so you are less likely to lose your work if there is an internet hiccup). Under the *compounds* tab, type in the names of the 6 components of a BTEX mixture, each on its own line. Press the *solve* button to see Restek's suggestions for separation. In this simulator, the resolution of each peak listed is with its closest neighbor in either direction. Notice that "results were found on 6 stationary phases." Look through these six results. Which components of a BTEX mixture are the most difficult to separate?

Look up the boiling points of the 3 xylene isomers in a BTEX mixture, and record them in your ELN. (They are not identical.) Does this information explain your previous answer? Why / why not?

In the GC simulator, select the stationary phase that is most like the one used in the paper you selected. (Determining this similarity may take some internet research.) Set all of the simulator parameters as closely as possible to those in the paper. Note that you must click "Custom" in the Results tab to be able to make changes in the Control Parameters tab. Click "Vacuum" if trying to simulate a GCMS method, or "Atm" if any other detector was used. Make a table comparing reported retention times from the paper and the simulator predictions in the ELN Data & Obs. In a third column, calculate the absolute value of the difference between them, $|\Delta t_r|$. How accurate is the simulator, in terms of retention order?

What is the average $|\Delta t_r|$ error and standard deviation of the simulator's predictions? (You may use Python, Excel, or a calculator to answer this last question.)

There is an ongoing shortage of helium, which has caused some GC operators to switch to other mobile phase "carrier gases". Planning how to switch over a method from one carrier gas to another is one of the purposes of this GC simulator. Make a second table and record the simulator's retention times and resolution for each peak with helium, leaving room for a matching dataset using nitrogen as the carrier gas. Switch the carrier gas from helium to nitrogen in the simulator. Do you have to raise or lower the column flow to get the retention times to where they were before?

Can you get the retention times to be essentially the same with nitrogen as they were with helium? How close can you match them? Include this data and the optimized nitrogen gas pressure in your ELN.

Once you change the pressure in this way, is the resolution of the chromatogram using nitrogen comparable to the resolution in the original one with helium? Or are there significant differences?

Given that a tank of ultra-high-purity nitrogen is much cheaper than a tank of UHP helium, which carrier gas would you recommend for this separation?

Lab exercise part 2: Optimizing gradient separations

For this part, we will use the **simplex optimization** technique for optimizing the GC separation of a fragrance sample with 10 components. To set up the simulator, select "Compounds," and then "Search by phase." Choose the Stabilwax stationary phase, then select the "flavors and fragrances" library. From the long list of scent molecules, select (-)-isopulegol, (*E*)-ocimene, ~~(*R*)-carvone,~~ α-humulene, α-phellandrene, α-terpinene, α-terpineol, β-caryophyllene, β-myrcene, and citral. (Citral has two isomers which are automatically included when you select it.) Click "solve."

Select the Conditions tab, and in the Results area at the bottom, click "Custom." (This allows you to change the Column and Control Parameters.) Under Carrier Gas, select Helium. Under Control Parameters, switch from vacuum to Atm. The outlet pressure should be 1 atm, or 14.7 psi. You are now simulating a GC instrument with a detector that does not pull a vacuum on the end of the column (in other words, *not* a mass spectrometer). Set up the following parameters:

Column: Stabilwax with dimensions of 30m length, 0.25 mm inner diameter, 0.5 μm film thickness
Column flow: 2.5 mL/min
Oven program: Ramps
Number of ramps: 1
Initial temp = 50 C
Initial hold time = 2 min
Ramp rate = 10 C/min
Final temp = 200 C
Control method: Constant Flow

As you make these changes, you will see that each one causes changes in the chromatogram, both in resolution and in retention order. I've highlighted 8 of them in red – the first three you normally get to choose only when buying a column, and the last five you can vary for a given run. Of course, with a

simulator, we get to vary all 8 for a given run. This is like having an infinite supply of columns to play with. You can see why Restek, a column manufacturer, maintains this simulator website.

Once you set all the parameters correctly, you will see that the first two peaks elute with only 0.4 resolution – clearly not what we want. Your goal in part 2 of the lab today is to optimize the separation by simultaneously varying FOUR parameters using the simplex optimization strategy. Your first task is to select 4 parameters from the 8 parameters highlighted in red on the previous page – at least one of them must involve the *oven program*:

1.
2.
3.
4.

Optimization setup. Next, we need to plan 5 experiments – representing the 1st four-dimensional simplex with its 5 corners – where the four parameters you've chosen are varied over 2-3 values within the "starting simplex range" listed in Table 1. DO NOT PICK ANY VALUES OUTSIDE OF THE RECOMMENDED STARTING RANGES. Note that the starting simplex ranges listed in the table cover 20 – 40% of the total available range for each parameter. This is a general experimental design principle, and consistent with the starting ranges in the 2-D simplex exercise we did in class. It gives the simplex some room to "walk around." (In Table 1, all 8 parameters are listed, because I don't know which 4 you will select.)

Table 1: Parameters for Optimization

Parameter	**Total range available**	**Starting simplex range**
Column flow (mL/min)	0 – 4	1.5 – 2.5
Column length (m)	10 – 60	15 – 30
Column inner diameter (mm)	0.1 – 0.53	0.15 – 0.25
Stationary phase film thickness (μm)	0.1 – 2	0.25 – 1
Initial oven temperature (C)	-20 to 200	40 – 80
Initial temperature hold time (min)	0 – 10	1 – 4
Temperature ramp rate (C/min)	2 – 50	10 – 20
Final oven temperature (C)	150 – 260	180 – 220

Build a table like the sample one below in your ELN, identifying the four parameters you've selected. (Your four parameters cannot be the same four as in the sample table.) The Expt #1 row should match the current settings on the EZGC simulator -- the simulator is currently displaying the results of Experiment 1! Then, plan Experiments 2, 3, 4, and 5, according to the following guidelines:

- No two experiments should be identical.

- For each of your 4 selected parameters, pick 2-3 values spanning the starting simplex range
- Distribute these values **randomly** to experiments 2, 3, 4 and 5! (Don't distribute them all low to high, for example.)

Here's what my Table 2 looked like when I finished the plan for the initial simplex set of experiments. I colored the conditions blue (for high), black (for middle) and red (for low) for each parameter to help me see that they were randomly distributed:

Table 2: Simplex optimization summary (example)

Vertex # (Expt #)	**Parameter 1 (units)**	**Parameter 2 (units)**	**Parameter 3 (units)**	**Parameter 4 (units)**	**Response**	**Vertices retained after expt.**
1	30	0.25	0.5	2.5		
2	20	0.25	0.25	2		
3	15	0.18	0.25	2.5		
4	20	0.25	0.5	1.5		
5	20	0.15	1	2		
6						

The next step is to decide on the **definition of the response** – this is a number that we'll be trying to maximize (or minimize). As you know, we want all peaks to be resolved with $R = 1.5$ (or greater), *and* we want the last peak's retention time to be as short as possible. We therefore need to combine these goals into one numerical summary. Your group is free to propose your own definition of response – and whether it needs to be maximized or minimized! Here is one possible idea:

$$Response = \frac{(\text{minimum } R)^2}{t_r \text{ of last peak}}$$

The bigger the minimum resolution (between the two closest peaks), and the smaller the last peak retention time, the larger *Response* will be. Thus the optimization goal would be to maximize *Response*. "Minimum *R*" is squared so that it will have a bigger impact on the optimization.

Our group defines the response as follows:

Response =

Will this response need to be maximized or minimized?

At this point, flag down Dr. D, and have him OK the first 5 experiments and the definition of response that you plan to use. Once you've received the OK, run experiments 2-5 on the GC simulator, calculate the response in each case, and add these values to your table in the ELN. Here's what my table looked like after my first five experiments ran on the simulator:

Table 3: Simplex optimization summary (example)

Vertex # (Expt #)	Parameter 1 (units)	Parameter 2 (units)	Parameter 3 (units)	Parameter 4 (units)	Response	Vertices retained after expt.
1	30	0.25	0.5	2.5	$0.4^2/13.91 = 0.0115$	
2	20	0.25	0.25	2	$1.7^2/10.85 = 0.2664$	
3	15	0.18	0.25	2.5	$2.2^2/9.82 = 0.4929$	
4	20	0.25	0.5	1.5	$0^2/13.11 = 0$	
5	20	0.15	1	2	$0.5^2/15.28 = 0.0164$	
6						

Simplex walking. Now that you've run the 5 experiments defining the simplex, it's time to *let the simplex method decide what to do next*. First, reject the experiment that gave the worst response – in my example, expt 4. (This means you retain the other 4 vertices or experiments – 1,2,3,5 in my example.) Then, calculate the Experiment 6 value for each parameter, using the familiar rule:

Next parameter value = 2(average of retained experiment values) – (rejected experiment value)*

For example, based on my above table, the next parameter 1 value I should try will be

2*(30+20+15+20)/4 – 20 = 22.5 m

(If you want to do these calculations in Excel or Python to speed things up, you may do so.) I've completed my plan for Experiment 6 in the table below, using the simplex rules. Note that my experiment 4 gave the worst results, when Parameter 2 was high and Parameter 4 was low. The simplex rules tell me to do my next experiment with a low value for Parameter 2 and a high value for Parameter 4!

Table 4: Simplex optimization summary (example)

Vertex # (Expt #)	Parameter 1 (units)	Parameter 2 (units)	Parameter 3 (units)	Parameter 4 (units)	Response	Vertices retained after expts.
1	30	0.25	0.5	2.5	$0.4^2/13.91 = 0.0115$	na
2	20	0.25	0.25	2	$1.7^2/10.85 = 0.2664$	na
3	15	0.18	0.25	2.5	$2.2^2/9.82 = 0.4929$	na
4	20	0.25	0.5	1.5	$0^2/13.11 = 0$	na
5	20	0.15	1	2	$0.5^2/15.28 = 0.0164$	1,2,3,5
6	22.5	0.165	0.5	3		

Your group's task is to take your optimization out to Experiment 12. Extend your summary table to show the conditions for each experiment, the response, and the vertices retained after each experiment. Don't forget about rules 3 and 4! After 12 experiments, highlight the optimal set of conditions that you have found using the technique of simplex optimization, like in my example below.

IMPOTANT NOTE: In my optimization, I got the best results by using a **modified rule 4**: if the simplex rules suggested a negative (impossible) value for Parameter 3, I interpreted this as a request to use a smaller value than before. So, rather than setting the result to zero, I substituted the negative value with the smallest value that is commercially available (0.1 for this parameter). This modification allowed my simplex to walk uphill instead of floundering around.

Here is my example table after 12 experiments. Notice that I used a highlighter to keep track of my current simplex vertices:

Table 5: Simplex optimization summary (completed example)

Vertex # (Expt #)	**Parameter 1 (units)**	**Parameter 2 (units)**	**Parameter 3 (units**	**Parameter 4 (units)**	**Response**	**Vertices retained after expt.**
1	30	0.25	0.50	2.5	$0.4^2/13.91 = 0.0115$	na
2	20	0.25	0.25	2.0	$1.7^2/10.85 = 0.2664$	na
3 best response	15	0.18	0.25	2.5	$2.2^2/9.82 = 0.4929$	na
4	20	0.25	0.50	1.5	$0^2/13.11 = 0$	na
5	20	0.15	1.00	2.0	$0.5^2/15.28 = 0.0164$	1,2,3,5 (reject4)
6	22.5	0.17	0.50	3.0	$0.3^2/12.94 = 0.0070$	2,3,5,6 (reject1*)
7	8.8	0.10	0.50	2.3	$1.6^2/10.8 = 0.2370$	2,3,5,7 (reject6)
8	9.4	0.20	0.50	1.4	$1.6^2/10.64 = 0.2406$	2,3,7,8 (reject5)
9 fastest baseline separation	6.6	0.20	0.10#	2.1	$1.5^2/6.14 = 0.3664$	2,3,8,9 (reject7)
10	16.7	0.30	0.10	1.7	$1.6^2/8.59 = 0.2980$	2,3,9,10 (reject8)
11	19.8	0.3	0.10#	2.8	$1.8^2/8.16 = 0.3971$	3,9,10,11 (reject2)
12	9	0.2	0.10#	2.5	$1.6^2/6.19 = 0.4136$	

*rule 3 invoked: if new vertex point is worst, reject 2nd worst point to keep simplex from backtracking.
#modified rule 4 invoked: simplex rules suggested a negative value for Parameter 3, so I substituted the lowest one available from Restek. Yellow highlights indicate 5 vertices of final position of simplex.

In 7 steps (Expts. 6 – 12), my simplex optimization was not able to find a higher response than Expt. 3 (a lucky guess!), but it *was* able to reduce the time to get a baseline separation by 3.5 minutes! Summarize what your simplex was able to achieve in its 7 steps, as part of a 3-4 sentence conclusion (R&A).

Lab exercise part 3: Building a model using multivariate regression

In Module 5a, once you determined the mathematical relationship between phi (the mobile phase mixture) and k (the retention factor), you could create your own model that predicted how the LC separation would behave for *any* mobile phase mixture, independent of the online simulator. In Module 5b today, you've used simplex optimization to try to find the best chromatographic separation while varying four GC parameters. But we haven't yet created a model, and after parts 1 and 2 of lab today you may not yet have a clear feeling for which of your GC parameters were most influential. So, creating a model is the next step – a model that will show how our separation depends on your four parameters, alone and in combination. We'll learn how to build such a model in class on Thursday, which will be led by experimental design guru Professor Lenny Perry from the School of Engineering.

The last task in lab today is to collect the input data that we'll need for building this model, so you have it ready to go on Thursday morning. We'll use the following process:

A. For each of your four parameters choose two values ("levels") with your lab group.
   1. Parameter 1_____. High = _____ Low = _____
   2. Parameter 2. _____. High = _____ Low = _____
   3. Parameter 3. _____. High = _____ Low = _____
   4. Parameter 4. _____. High = _____ Low = _____

B. Now, splitting up the work with your lab group, you'll need to run a simulation with every possible combinations of High and Low values for the four parameters, and record the results. This experimental design is known as "factorial," because the number of experiments required is (# parameters)$^{\#levels}$, or $2^4 = 16$ in this case. Use the table below to ensure you get all 16 combinations and no repeats:

Parameter 1	Parameter 2	Parameter 3	Parameter 4	Response
high	high	high	high	
low	high	high	high	
high	low	high	high	
high	high	low	high	
high	high	high	low	
low	low	high	high	
low	high	low	high	
low	high	high	low	
high	low	low	high	
high	low	high	low	
high	high	low	low	
low	low	low	high	
low	low	high	low	
low	high	low	low	
high	low	low	low	
low	low	low	low	