# LAB MANUAL & PROCEDURE
# $C_4H_3F_7O$ (1,1,2,2-Tetrafluoroethyl 2,2,2-trifluoroethyl ether, TFTFE)

Aziell Paul S. Limbo, Advisor: Dr. Karine Le Bris
*California State University, Los Angeles, Department of Physics and Astronomy, Los Angeles, CA*
(Dated: December 2025)

**Contents**



## I. INSTALLATION

Install the following (this is assuming `Ubuntu` is already installed, if needed, install `Ubuntu`. There are many different ways to install it if needed at all):

- **Avogadro** This program is used for "drawing" your molecule. Can be done using this link `https://avogadro.cc/`
- **ORCA** This software is the real meat of it all. Most calculations will be done here. I installed it onto `Ubuntu` using this tutorial: `https://spoken-tutorial.org/watch/ORCA+-+Computational+Chemistry/Installation+of+ORCA+on+Linux/English/`

  **NOTE:** I admit it's a little dated, but the instructions should be similar to the time of writing this manual. I have `ORCA 6.1.0` installed on my machine and used the very same tutorial. Let's hope not much has changed as time goes on.

  **GOOD PRACTICE:** It's also not a bad idea to read the `ORCA` documentation and run your first calculation at this point. Documentation can be found here: `https://www.faccts.de/docs/orca/6.0/manual/`. For my first run, I read the entire beginning up until I made my first `Hello Water` calculation as described. Documentation also tells you the "official" way of installing `ORCA`, but it's not easy to follow if you're not too sure what you're doing.

I am on Windows, and access `Ubuntu` through the `cmd` prompt. As for the text editor, I use `vim`, but any editor can be used.

## II. BUILDING MOLECULE & INPUT FILE

TFTFE has three conformers (I only know this from this source: `https://pubchem.ncbi.nlm.nih.gov/compound/164596`). This means three separate files will need to be created via Avogadro. In turn, we will run `ORCA` three times separately for each conformer.

Learn how to draw molecules in Avogadro using this online tutorial: `https://www.youtube.com/watch?v=zLBz8WiXbAs`. Personally, I uncheck the "Adjust Hydrogens" setting so I have complete control of what I am inputting - just don't forget your hydrogen atoms where needed!

Once a rough 3D drawing is made, the geometry of the molecule needs to be optimized. It's unlikely you've drawn it the exact mathematical way if you used just your mouse to draw your molecule. Go to:

```
Extensions >> Optimize Geometry
```

It should now look like a legit molecule with mathematically correct angles of bonds and atoms. Assuming everything looks correct, feel free to use this visual in final papers and posters by exporting it as a `png` (I totally did this in Fig. 1).

**NOTE** DEFINITELY save this file! Avogadro saves it's own files using the `.cml` extension.

Now it's time to set everything up for `ORCA`. Go to:

```
Extensions >> Orca >> Generate Orca Input...
```

Hopefully by this point you've learned a thing or two about how to configure your calculations from the `Hello Water` example described in the documentation for `ORCA`. This time you edit the input file within the `Orca Input Parameters` pop-up window instead of manually writing an input file as you may have done with the `Hello Water` example using some arbitrary text editor. Around line 4 (where the exclamation point is) is where you tell `ORCA` what to calculate and more importantly *how* to calculate. Anything following a # symbol is a comment. For my first example, I used this input:

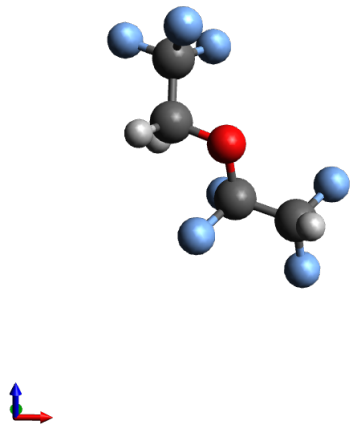

FIG. 1: An exported `png` file of what I am currently calling the "first conformer" of TFTFE

```
.
.
.
! B3LYP def2-SVP OPT FREQ
.
.
.
```

**NOTE:** More of a confession really. At the time of writing this, I don't fully understand the various DFT and basis set calculations yet. I have arbitrarily chosen the `B3LYP` method as it was referenced in my prior reading into doing any kind of calculations. `def2-SVP` was arbitrarily chosen at it is one (of the many) method(s) used in Dr. Le Bris's previously published papers. Hopefully this note gets deleted very soon so I may say otherwise.

The `OPT` and `FREQ` directions are taken as we are interested in furthering optimizing the geometry of the molecule (apparently Avogadro does not do this fully, think of it as an added step to ensure all calculations are accurate) as well as calculating the various frequencies of the molecule.

Around line 6 (where the asterisk, *, is) shows the xyz coordinates, charge (usually zero), and multiplicity (usually 1) - in that order. I've left this setting for all conformers (for now):

```
.
.
.
* xyz 0 1
.
.
.
```

Now generate the input file, and ensure the file is saved as an `.inp` file. I recommend saving it as (or something similar):

```
[molecule name].inp
```

Now it's time to run `ORCA`! Now run your input file as you may have done in the `Hello Water` example. For me, I run this command in `Ubuntu`:

```
orca [molecule file name].inp > [your molecule file name].out
```

It took me 30 minutes to an hour for each conformer. So it may take a while!

## III. EXTRACTING RAW SPECTRAL DATA FROM ORCA OUTPUT

Ensure that the IR spectrum data is available in the `ORCA` output file. It should have a header followed by data like this:

```
.
.
.
-----------
IR SPECTRUM
-----------

 Mode    freq    eps    Int   ...
        cm**-1 ...
------------------------------...
[DATA]
.
.
.
```

Now the goal here is to generate a spectra graph for your molecule. Ideally we want to extract this data into a 2 column format so we can plot it. This can be done by hand, but I've made a `MATLAB` program that accomplishes this by scanning and extracting the data and outputs it into a `.dat` file. The program is entitled `IR_DATA_EXTRACT.m`. The source code is provided in it's own section.

For plotting the data. I used `xmgrace`. However, any plotting software can be used. The output of this is in Fig. 2

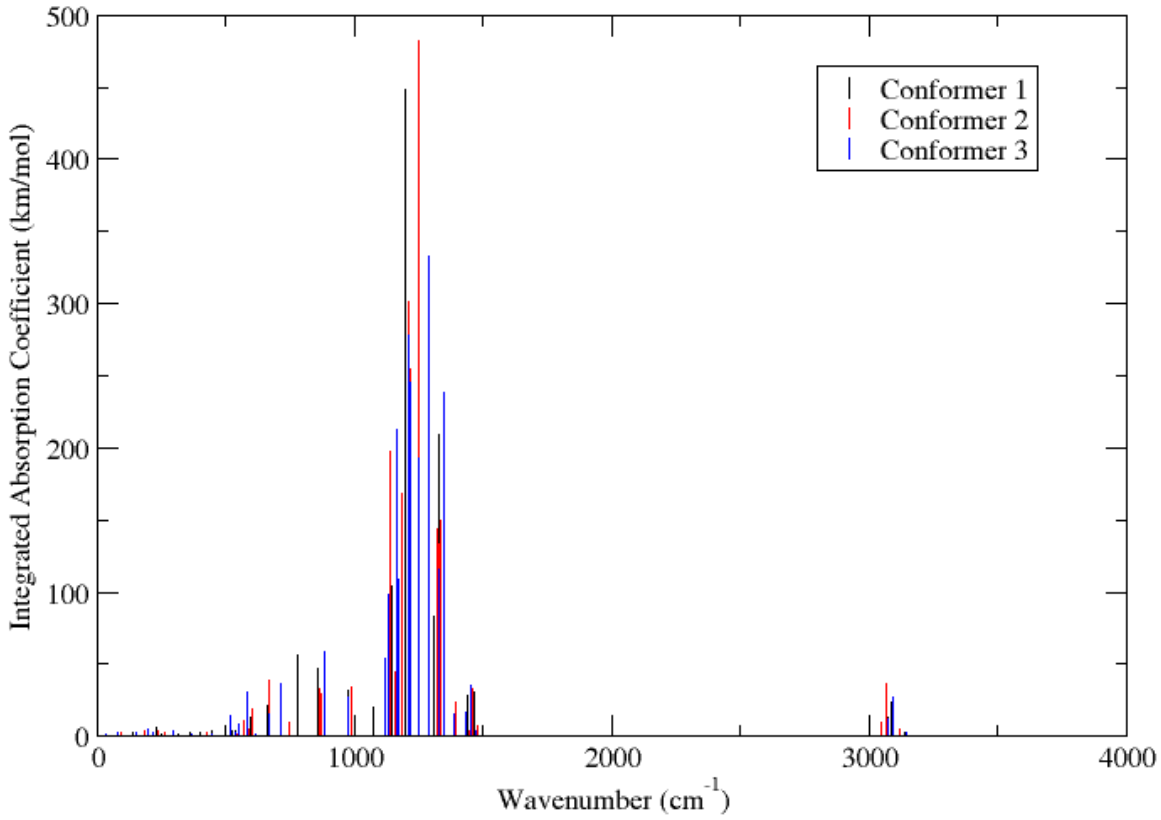


FIG. 2: Raw spectral graph of TFTFE produced by `xmgrace`. Data is directly from `ORCA`.

## IV. CALCULATING POPULATION PERCENTAGE

Now we wish to find the population percentage of each conformer. It's a simple calculation that needs no program to do. Still not a bad idea to write a program for one if there are many conformers For a given conformer, $i$, the population percentage is:

$$\text{Population}\%_i = \frac{g_i \cdot exp(-\frac{G_i - G_{min}}{kT})}{z} \tag{1}$$

where $g_i$ is the degeneracy value for the $i$th conformer, and $z$ is the partition function:

$$z = \sum_i g_i exp(-\frac{G_i - G_{min}}{kT}) \tag{2}$$

$k$ is the Boltzmann constant and $G_{min}$ is the lowest Gibbs free energy from each conformer, and this value is provided by `ORCA`. $T$ is temperature - and we assume room temperature for now at the value 298.15K. So each $G_i$ corresponds to to the Gibbs free energy for the $i$th conformer. For TFTFE there are three conformers, so there are three Gibbs free energies. To ensure the population percentage calculation is correct, the sum of these percentages should of course be equal to 1. All results are displayed in Table I

**NOTE** `ORCA` gives each $G$ value in Hartrees instead of Joules! So the conversion $1He = 4.359748199E - 18J$ must be applied.

| Conformer # | $g_i$ | Gibbs Free Energy ($J$) | Population Percentage |
|---|---|---|---|
| 1 | 1 | $-4.04248581 \times 10^{-15}$ | 83.2358451254% |
| 2 | 1 | $-4.04245942 \times 10^{-15}$ | 0.136797200712% |
| 3 | 1 | $-4.04247918 \times 10^{-15}$ | 16.6273576739% |

TABLE I: Table showing the Gibbs Free Energy in Joules for each conformer. $G_{min} = G_1$

IR_DATA_EXTRACT.m

```
%initialize storage
wavenumber=[];
intensity=[];


%open file
fid=fopen('TFTFE_C3.out'); %REPLACE WITH NAME OF INPUT FILE

%what to look for
pattern='IR SPECTRUM';

%search for it and extract
while ~feof(fid)
    line=fgetl(fid);
    if contains(line,pattern)
        for i=1:5
            line=fgetl(fid);
        end
        data_count=0;
        while ~feof(fid)
            line=fgetl(fid);
            if isempty(line)
                break;
            end
            line_clean=strrep(line, ':',' ');
            nums=sscanf(line_clean,'%f');
            if length(nums)>4
                wavenumber(end+1)=nums(2);
                intensity(end+1)=nums(4);
            end
        end
        break;
    end
end

%close the input file for good programming practice <3
fclose(fid);

%write data to an output file
fid_out=fopen('TFTFE_C3_IRDATA.dat','w'); %REPLACE WITH NAME OF OUTPUT
for i=1:length(wavenumber)
    fprintf(fid_out,'%.4f\t%.6f\n',wavenumber(i),intensity(i));
end

%close the output file for good programming practice <3
```

```
fclose(fid_out);
```