Seahorse Analysis

Mitochondrial Stress Test

Abrar Al-Shaer

Raz Shaikh Lab

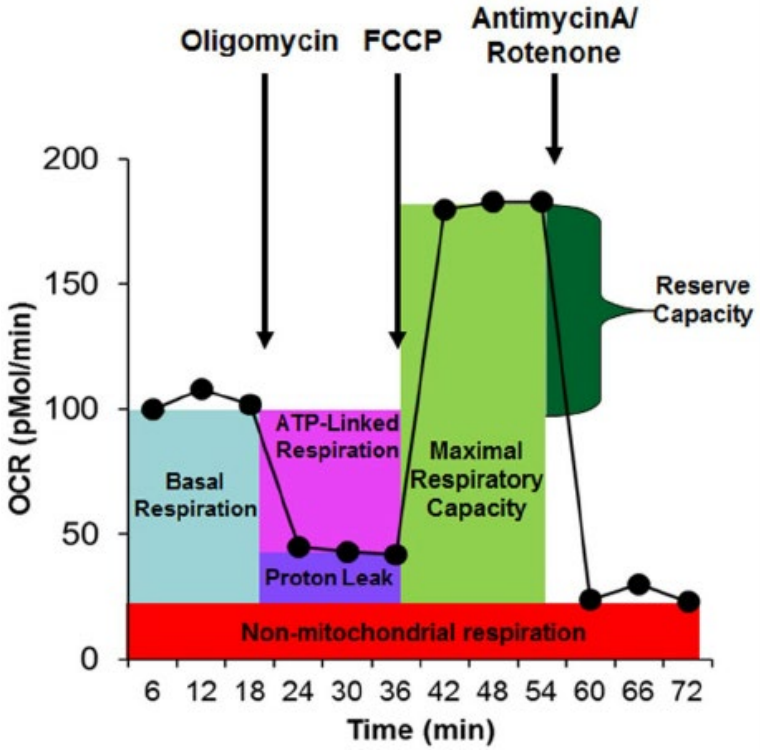
12/11/2018

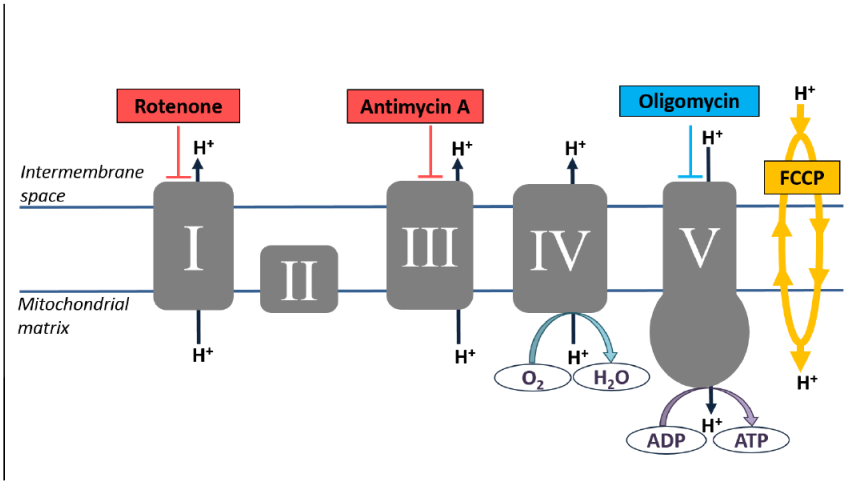
I. Introduction

a. The Seahorse Mitochondrial Stress Test is a method to measure mitochondrial function in any cell type. The expected curve and defined terminology are below:

Seahorse (gives you a read out of which cells are more metabolically active):

- **OCR** – oxygen consumption rate → measures mitochondrial respiration
- **ECAR** – extracellular acidification rate → measures proton excretion (more for glycolysis)





- **Oligomycin** → inhibits ATP synthase
- **FCCP** → proton uncoupler
- **Antimycin A** (inhibits complex III) and **Rotenone** (inhibits complex I).

b. Parameter Definitions

• **Basal respiration:** Oxygen consumption used to meet cellular ATP demand and resulting from mitochondrial proton leak. Shows energetic demand of the cell under baseline conditions.

• **ATP production:** The decrease in oxygen consumption rate upon injection of the ATP synthase inhibitor oligomycin represents the portion of basal respiration that was being used to drive ATP

production. Shows ATP produced by the mitochondria that contributes to meeting the energetic needs of the cell.

- **H+ (Proton) leak:** Remaining basal respiration not coupled to ATP production. Proton leak can be a sign of mitochondrial damage or can be used as a mechanism to regulate the mitochondrial ATP production.
- **Maximal respiration:** The maximal oxygen consumption rate attained by adding the uncoupler FCCP. FCCP mimics a physiological "energy demand" by stimulating the respiratory chain to operate at maximum capacity, which causes rapid oxidation of substrates (sugars, fats, amino acids) to meet this metabolic challenge. Shows the maximum rate of respiration that the cell can achieve.
- **Spare respiratory capacity:** This measurement indicates the capability of the cell to respond to an energetic demand as well as how closely the cell is to respiring to its theoretical maximum. The cell's ability to respond to demand can be an indicator of cell fitness or flexibility.
- **Nonmitochondrial respiration:** Oxygen consumption that persists due to a subset of cellular enzymes that continue to consume oxygen after rotenone and antimycin A addition. This is important for getting an accurate measure of mitochondrial respiration.

II. Goal (Expected Result)

You will have an outputted OCR and ECAR file with the measurements at multiple timepoint during the assay. Running calculations from these results will give you information on the mitochondria's functional capacity.

III. Methodology & Tools

a. Installing Packages/Software

The following tools used for this analysis are:

- **Wave Seahorse Software**
- **GraphPad Prism**
- **R & R Studio**
- **Python 2.7 & Python Pandas**

b. Obtaining & Formatting Data Files

i. Open the seahorse file outputted from the machine in the Wave software and click on Add → Overview to see the OCR & ECAR curves.
ii. While you are looking at the OCR curve, click on wells in the plate layout to see if any of the wells failed (meaning the curve doesn't exist for that well and the line plateaus).
iii. Click on all the failed wells you wish to exclude from the analysis (they will be greyed out).
iv. Next, to export the data click on Export, then GraphPad Prism.
v. Copy and paste the OCR & ECAR data from GraphPad into Excel as a transposed matrix then save it as a CSV file.
    1. We are exporting the data to GraphPad then copying it to excel instead of exporting to Excel directly because the output to GraphPad provides you with averages of each well for each timepoint – the Excel output contains raw data we do not need.
vi. The excel CSV file should now look like this:

Time (minutes)	1.413564	7.986783	14.60099	21.27969	27.8488	34.43665	41.07356	47.68047	54.26819	60.91076	67.49809	74.09724
B cell Lean 1	49.57615	47.54439	45.74795	37.4895	21.71886	17.41249	104.8447	93.198	80.84441	10.25219	10.19917	9.8429
B cell Lean 1	43.90448	39.38402	37.84238	24.46187	15.26132	12.57663	82.93594	74.61983	71.76517	8.785859	8.003402	9.159294
B cell Lean 1	44.76375	40.52185	39.39368	36.06548	24.72655	15.6071	85.2989	77.084	71.85423	9.542321	9.139562	9.41633
B cell Lean 2	50.96995	47.34839	45.73352	38.763	27.44122	19.60514	96.79631	86.25178	81.46457	10.04888	10.17928	9.986348
B cell Lean 2	51.01281	46.54608	43.50277	36.94446	26.8708	19.84905	92.7298	85.13805	81.0997	9.057863	8.754697	9.395933
B cell Lean 2	52.30363	50.88255	47.68903	42.62968	30.65907	19.89789	99.78818	90.47228	85.46352	11.05722	10.69626	10.98713
B cell Lean 3	65.50074	62.60739	59.28943	55.3773	39.90716	28.49746	128.0628	119.3665	113.3544	13.7951	13.37946	13.99678
B cell Lean 3	64.31208	59.49124	57.17996	50.03301	31.51518	22.20983	132.3583	121.3854	114.5883	10.78081	11.74585	11.84775
B cell Lean 3	68.12185	63.64025	61.25235	61.70935	59.19243	53.51549	124.094	120.6686	118.7597	13.15446	13.04944	13.81353
B cell Lean 4	50.58035	48.17221	45.20075	42.21553	34.13883	23.92357	92.47896	83.13595	78.1644	12.96525	12.53031	12.39374

vii. Data Pre-processing steps for R & python scripts:
   1. If you wish to exclude entire rows (wells) from the analysis be sure to remove them. DO NOT remove individual values yet (reference step 2 below).
   2. If you wish to exclude any individual row value from the analysis, replace the value with "NaN". Python will skip NaN values and exclude them from analysis.
      a. If each column does not contain same number of rows, the program will fail.
   3. You **MUST** replace all negative OCR & ECAR values with NaN in the CSV file!
   4. Make sure to label each column with the corresponding biological replicate (ex: B cell Mouse 1)
   5. **ALL BIOLOGICAL REPLICATES MUST HAVE THE SAME UNIQUE PREFIX** (ex: B cell Mouse 1, B cell Mouse 2, CD4 Mouse 1, CD4 Mouse 2, etc...) & prefixes must **NOT** overlap between biological groups.
      a. The biological group names must be unique from each other, from example, you cannot have B cell HF and B cell HF_SPM because both contain the same prefix 'B cell HF' and they are different biological groups.
      b. That is why you should change it to B cell HF and B cell SPM_HF to make sure all prefixes are unique. Or else the program will assign that sample to the wrong group. ***Make sure the ECAR & OCR Group names match***

c. **Calculating Outliers**
   i. To analyze whether a technical replicate is an outlier, run the **seahorse_outlier_analysis.R** program, or calculate outliers with the 1.5*IQR method.
   ii. In the seahorse_outlier_analysis.R program, the outliers will be outputted into a matrix datafile where you will have the biological group, timepoint, and outlier OCR or ECAR value outputted to a file (example below).

c("B cell Lean 2"	"X60.91076"	"27.012544")
c("B cell Lean 2"	"X67.498093"	"29.071465")
c("B cell Lean 2"	"X74.097237"	"29.478494")
c("B cell Lean 3"	"X21.279686"	"22.35429")

   iii. Using the output above, the R program will replace all outliers with NaN. This segment of the code is under the comment: "**#replacing outliers with NaN**".
      1. **Note:** There are usually never ECAR outliers
   iv. Open the new OCR or ECAR file that contains the replaced outliers and look to see if there are consecutive lists of NaNs that span entire timepoints (every 3rd column spans a timepoint – reference the "Experimental Group #1" image below). If a row has NaNs that span an entire timepoint – meaning there are no values for an entire timepoint – then **delete that row**.

d. Calculating Parameters
   i. The python program runs the equations to calculate different parameter outputs for the Mito Stress Test. The equations used for the calculations are below:

Parameter Value	Equation
Non-mitochondrial Oxygen Consumption	Median rate measurement after Rotenone/antimycin A injection
Basal Respiration	(Last rate measurement before first injection) – (Non-Mitochondrial Respiration Rate)
Maximal Respiration	(Maximum rate measurement after FCCP injection) – (Non-Mitochondrial Respiration)
H+ (Proton) Leak	(Minimum rate measurement after Oligomycin injection) – (Non-Mitochondrial Respiration
ATP Production	(Last rate measurement before Oligomycin injection) – (Minimum rate measurement after Oligomycin injection)
Spare Respiratory Capacity	(Maximal Respiration) – (Basal Respiration)
Spare Respiratory Capacity as a %	(Maximal Respiration) / (Basal Respiration) × 100
Acute Response	(Last rate measurement before oligomycin Injection) – (Last rate measurement before acute injection)
Coupling Efficiency	ATP Production Rate) / (Basal Respiration Rate) × 100

	Experimental Group #1						
	Measurement	Assay Well - A1 OCR (pmol/min)	Assay Well - A2 OCR (pmol/min)	Assay Well - A3 OCR (pmol/min)	Assay Well - A4 OCR (pmol/min)	Average OCR	
Baseline OCR	1	170.02	172.80	169.96	175.99	172.19	
	2	165.36	163.62	167.00	166.42	165.60	
	3	160.50	158.25	159.44	161.75	159.98	<< Last rate measurement before 1st injection
Injection 1 - Oligomycin	4	65.44	53.05	61.53	59.22	59.81	
	5	61.80	49.44	59.62	56.94	56.95	
	6	61.93	49.55	59.06	56.04	56.65	<< Minimum rate measurement after Oligo injection
Injection 2 - FCCP	7	319.58	310.94	312.59	315.98	314.77	
	8	327.95	320.32	325.77	330.73	326.19	<< Maximum rate measurement after FCCP Injection
	9	297.17	292.30	301.35	299.68	297.63	
Injection 3 - Rot/AA	10	51.77	34.62	44.99	39.45	42.71	
	11	49.28	33.24	44.13	39.05	41.42	
	12	47.03	31.80	42.66	39.20	40.17	<< Minimum after Rot/AA Injection (Non-Mitochondrial Oxygen Consumption)

We use median rate

ii. When running the python program (**seahorse_analysis.py**), make sure to correctly list the biological group names as described in step ***vii***. Make sure to also change the input and output file names. The output file names are located on lines 111 and 123-125.
    1. Open bash & run: **python seahorse_analysis.py**. You will get 4 output files, we will use the output file from line 111 (first csv output) as input for our R graphing script.

e. Statistical Analysis & Graphing

i. Use the **seahorse_comboPlots.R** script to generate statistics & graph the results.
    1. If you have many biological groups and would like to run a particular set at a time, then run the command on line 21 under the comment (**#filter by group**). There you can exclude particular groups from the dataframe and only analyze a few at a time.

ii. You will get a graph output like the one below:
*The assumption of normality is checked in the code with the Shapiro-Wilks test to decide whether to use a T-test or Wilcoxon test for statistical comparisons.