**GitHub\_readme.docx:** A Microsoft Word file designed to help navigate other files provided as well as to outline the general file structure and barebones workflow used with the p-ath780 model to make model implementation easier.

**p-athSeed/p-ath780Seed.txt:** This file is a text file which contains the seed tissue model of p-ath780. This model contains some label reactions (reaction identifier includes the word “label”) which helps identify pathways to which blocks of reactions belong, allowing greater organization of the file. These reactions are sorted into a reaction subset which is not allowed to carry flux by the convert.py files, and therefore these lines essentially act as comment lines.

**p-athLeaf/p-ath780Leaf.txt:** This file is a text file which contains the leaf tissue model of p-ath780. This model contains some label reactions (reaction identifier includes the word “label”) which helps identify pathways to which blocks of reactions belong, allowing greater organization of the file. These reactions are sorted into a reaction subset which is not allowed to carry flux by the convert.py files, and therefore these lines essentially act as comment lines.

**p-athRoot/p-ath780Root.txt:** This file is a text file which contains the root tissue model of p-ath780. This model contains some label reactions (reaction identifier includes the word “label”) which helps identify pathways to which blocks of reactions belong, allowing greater organization of the file. These reactions are sorted into a reaction subset which is not allowed to carry flux by the convert.py files, and therefore these lines essentially act as comment lines.

**p-athStem/p-ath780Stem.txt:** This file is a text file which contains the stem tissue model of p-ath780. This model contains some label reactions (reaction identifier includes the word “label”) which helps identify pathways to which blocks of reactions belong, allowing greater organization of the file. These reactions are sorted into a reaction subset which is not allowed to carry flux by the convert.py files, and therefore these lines essentially act as comment lines.

**p-ath780.gms:** This file is the GAMS code for the p-ath780 model itself. This uses the outputs of various other codes included in this repository, and therefore users should consult the workflow described later in this text to ensure that this code functions correctly.

**p-ath[Tissue]/convert.py:** This is an executable Python code which takes the input of a model (such as p-athSeed/p-ath780Seed.txt) and outputs a number of files which can be read by GAMs code. This code has been modified for each [Tissue], and therefore should be able to be run by end users with no further modifications.

**makeGrowthInputs.pl:** This is an executable Perl code which takes the results of converting each tissue model file using convert.py and creates some of the necessary inputs for the p-ath780.gms code.

**growthSpecsNames.txt:** This is a text file which contains a list of the names of parameters used to defined p-ath780 model growth. Examples of the named parameters include tissue mass ratios, rates of nutrient uptake from seed stores, and rates of change for tissue mass fractions among other input parameters. Note that due to how GAMS works, this just contains the names for these parameters which will be referred to in code, the actual values are contained in growthSpecs.txt.

**growthSpecs.txt:** This is a text file which contains the actual specifications (values) used for growing by the p-ath780 model.

**timepointsH.txt:** This is a text file containing the list of time points to iterate over for each day, e.g. this contains each hour of the day, beginning at 0 and ending at 23. This is used in the definition of the set of hours of the day.

**timepoints.txt:** This is a text file containing a list of days to solve the model over, in this case from day 0 to day 61. This is used in the definition of the set of days over which the model simulates growth.

**timeData.txt:** This is a text file containing a list of data labels for much of the data saved at each time point (combination of day and hour) and reported on in the troubleshooting file as well as the output file.

**sunrise.txt:** This is a text file which lists the time at which the sun rises (or light is made available) each day. At present, light is made available at a default time of 0 hours into the day.

**sunset.txt:** This is a text file which lists the time at which the sun sets (or light is no longer made available) each day. At present, light is no longer available starting at 12 hours into the day.

**timeofday.txt:** This file basically converts the set time of day to a parameter of equal value. Necessary because mathematical operations cannot be performed on sets.

**RxnstoGene.pl:** This is an executable Perl code file which is used to automatically curate the Gene-Protein-Reaction (GPR) links for all tissue models using the KEGG API (advanced programming interface, rest.kegg.jp). Inside the documentation of the code is the instructions for adapting it to investigate the GPR links for each tissue. Generally, this file is named “RxnstoGenes.pl”. This file requires the LWP Perl package.

**PathGetRxnsComps.pl:** This file is an executable Perl code file (extension of “.pl”) which automatically generates the lists of reactions associated with various KEGG pathways. The KEGG pathways used are listed in Data S4.

**ModelPathComp.pl:** This file is an executable Perl code file which is used to automatically read the files created from Supplemental Files 7 to give counts of how many reactions a model has which belong to each of the pathways indicated by Text S7.

**Text\_S20.docx:** This file is a Microsoft Word file which contains all the calculations simplifications, and rational used for the determination of the function for whole-plant mass step with respect to time.

**Model\_Building\_and\_Calculations.xlsx:** This file is a Microsoft Excel file which store a wide variety of information concerning the p-ath780 model. This include the manually-curated GPR results for each tissue model, the calculations pertaining to the determination of the biomass equation for each tissue model, calculations for various parameters used in the p-ath780 model to incorporate literature data, and calculations pertaining to the diurnal storage and uptake of carbohydrates.

**p-ath780\_results.xlsx:** This is a Microsoft Excel file which contains the results of the p-ath780 model for various alternative objective functions. The sheet tabs indicate which alternative objective function the data corresponds to. The key is as follows:

lpe\_g\_g\_g\_X: linear photonic efficiency objective for the leaf, growth objective for other tissues. The number which replaces “X” indicates the run number.

g\_g\_g\_g: Growth objective for all tissues.

nlpe\_X: Non-linear photonic efficiency objective, the number which replaces the “X” denotes the run number.

g\_g\_fa\_g\_X: Fatty acid storage objective for the seed tissue, growth objective for all other, the number which replaces the “X” denotes the run number.

**pathList.csv:** This file is a comma separated values file. This file contains a list of 73 KEGG pathways for which to get the list of associated reactions. This file is referred to by code as “.csv”, and must have the proper name for the code to function properly. Generally referred to as “PathRxns.csv”.

**General File Structure and Workflow Used in Workflow:**

**Main Folder:** p-ath780

**Files:** ModelAnalysis.xlsx, makeGrowthInputs.pl, growthSpecsNames.txt, growthSpecs.txt, timepointsH.txt, timepoints.txt, timeData.txt, sunrise.txt, sunset.txt, timeofday.txt, p-ath780.gms, and Supplemental\_File\_10.docx

**Sub-Folder:** p-athLeaf

**Files:** p-ath780Leaf.txt, convert.py, pathList.csv, PathGetRxnsComps.pl, ModelPathComp.pl, RxnstoGenes.pl

**Workflow:**

1. Add all above files to folder
2. Run “convert.py” to generate files readable by and necessary for p-ath780.gms
3. If desired, run PathGetRxnsComps.pl and ModelPathComp.pl to get initial set of data behind the construction of Fig. 2.
4. Run RxnstoGenes.pl if desired to get initial GPR link information which is also used in Fig. 2.

**Sub-Folder:** p-athRoot

**Files:** p-ath780Root.txt, convert.py, pathList.csv, PathGetRxnsComps.pl, ModelPathComp.pl, RxnstoGenes.pl

**Workflow:**

1. Add all above files to folder
2. Run “convert.py” to generate files readable by and necessary for p-ath780.gms
3. If desired, run PathGetRxnsComps.pl and ModelPathComp.pl to get initial set of data behind the construction of Fig. 2.
4. Run RxnstoGenes.pl if desired to get initial GPR link information which is also used in Fig. 2.

**Sub-Folder:** p-athSeed

**Files:** p-ath780Seed.txt, convert.py, pathList.csv, PathGetRxnsComps.pl, ModelPathComp.pl, RxnstoGenes.pl

**Workflow:**

1. Add all above files to folder
2. Run “convert.py” to generate files readable by and necessary for p-ath780.gms
3. If desired, run PathGetRxnsComps.pl and ModelPathComp.pl to get initial set of data behind the construction of Fig. 2.
4. Run RxnstoGenes.pl if desired to get initial GPR link information which is also used in Fig. 2.

**Sub-Folder:** p-athStem

**Files:** p-ath780Stem.txt, convert.py, pathList.csv, PathGetRxnsComps.pl, ModelPathComp.pl, RxnstoGenes.pl

**Workflow:**

1. Add all above files to folder
2. Run “convert.py” to generate files readable by and necessary for p-ath780.gms
3. If desired, run PathGetRxnsComps.pl and ModelPathComp.pl to get initial set of data behind the construction of Fig. 2.
4. Run RxnstoGenes.pl if desired to get initial GPR link information which is also used in Fig. 2.

Once the sub-folder operations are complete, return to the main folder.

In the main directory, run “makeGrowthInputs.pl”, this should create all necessary input files for p-ath780.gms which have not already been created.

Finally, the user should now be able to run p-ath780!