Pathway Tools / PythonCyc API Highlights

- PC = pythoncyc
- PC.select_organism(“meta” or “ecoli”) → pgdb
- pgdb.
    - all_direct_forms_of_protein(p: protein) → all direct subunits & containers of p
    - all_enzymes(t: type)
    - all_genetic_regulation_proteins(c: class) → all proteins involved in genetic regulation of c
    - all_modulators() → list of direct regulators
    - all_pathways(base: Bool) → list of pathways (base, or base + superpathways)
    - all_products_of_gene(g: Gene) → list of proteins encoded by g
    - all_reactions(t: all | metab_pathways | metab_smm | metab-all | enzyme | …)
    - all_substrates(rxns: list of reactions) → all substrates used in rxns
    - all_transcription_factors()
    - compartment[s]_of_rxn(rxn: Reaction) → compartment
    - compounds_of_pathway(p: Pathway) → all substrates of all reactions in p
    - deactivated_or_inhibited_by_compounds(cpds: list of compounds) → list of pathways affected
    - enzymes_of_gene(g) → list of enzymes
    - enzymes_of_pathway(p) → list of all enzymes
    - enzymes_of_reaction(r)
    - genes_of_pathway(p: Pathway) → list of genes coding for enzymes in the pathway
    - genes_of_protein[s](p)
    - genes_regulated_by_gene(g) → list of genes regulated by g via a transcription factor
    - genes_regulated_by_protein(p)
    - genes_regulating_gene(g)
    - get_predecessors(rxn, pwy) → list of reactions
    - get_slot_value(frame_id, slot_name), put_slot_value(frame_id, slot_name, val)
    - run_fba(file_name) → reactions + fluxes, biomass flux
        - Optimize - input file can optimize biomass, secretions, nutrients, etc.
        - Has **Gap-Filling** [with cost] - add reactions from MetaCyc to produce missing biomass
    - Process for FBA
        - Make a pgdb: pythoncyc.select_organism(base_organism_name)
        - Query and set_slot_value as needed
        - my_pgdb.save_pgdb()

Pathway Tools - Notes

- Compound - search by SMILES substructure (tutorial)

PyoFuel - Using Python and Pathway Tools to engineer synthetic Biofuel

SRI's Pathway Tools is a collection of biological modeling tools with databases of organism models, metabolic flux analysis with multiple gap-filling, and query and visualization tools. Pythoncyc is a Python programming interface to Pathway Tools.

In two earlier projects I had experimented with flux balance analysis on models of bacteria that had been modified with pathways to synthesize biofuel, and with wet-lab recombineering of the DHX35 gene using E.coli. The former was fun, quick, and easy; the latter was fun, but slow, and painful. So I wanted to use Python to script Pathway Tools, to help find candidate biofuel pathways across organisms, identify the corresponding gene-edits to engineer biofuel-friendly E.coli, and evaluate how effective each engineered organism might be using flux balance analysis -- all as a precursor to either more detailed modeling or wet-lab work.

PyoFuel is the resulting project. It is ongoing work, and my poster will report on the following using flowcharts, relevant Pythoncyc API calls, PyoFuel code snippets, and Pathway visualizations. The processing flow within PyoFuel is:

- Find candidate biofuel metabolites in MetaCyc, a multi-organism database
- Identify the pathways that produce those metabolites
- Filter out implausible pathways
- Generate a modified organism database to evaluate via FBA
- Run MetaFlux on the modified organism with suitable objectives and constraints
- Filter out those organisms if key flux numbers are poor
- Identify enzymes and corresponding genes for the modified pathways

I am currently a senior in high school. If accepted, I plan to open-source the current Jupyter notebook and pgdb databases, assuming I am permitted to do so. My info is at http://ashdza.github.io/.