

Reversible state machine

By design, A-ALPHA can react with n different sites (B-BETA, C-Gamma, ...), this will create n states in the first stage. A new site, B-ALPHA for example, will be created. This site will then be enabled to react again with one of m Z-OMEGA sites. This will create n*m distinct states. All states can be reverted using a reverse rule set.

```
DNA_sequence_4 = [
    'ONE', 'A', 'ALPHA',
    'TWO', 'beta', 'b',
    'THREE', 'gamma', 'c',
    'FOUR', 'omega', 'z',
    'FIVE', 'omega', 'z',
    'SIX', 'omega', 'z',
]
pairs_pool_4=[(['A', 'ALPHA'], ['B', 'BETA']), (['A', 'ALPHA'], ['C', 'GAMMA']),
               (['A', 'BETA'], ['Z', 'OMEGA']), (['A', 'GAMMA'], ['Z', 'OMEGA'])]
outcome_list_4=run_probability_calculation_and_plot_IS110(DNA_sequence_4,pairs_pool_4,numberofevents=3,plot=0,rank=1,print_o

[54] ✓ 1.9s Python

... ('ONE', 'A', 'BETA', 'two', 'alpha', 'b', 'THREE', 'gamma', 'c', 'FOUR', 'omega', 'z', 'FIVE', 'omega', 'z', 'SIX', 'omega', 'z')
('ONE', 'A', 'GAMMA', 'three', 'B', 'BETA', 'two', 'alpha', 'c', 'FOUR', 'omega', 'z', 'FIVE', 'omega', 'z', 'SIX', 'omega', 'z')
Stage 1 has 2 outcomes
('ONE', 'A', 'OMEGA', 'four', 'C', 'GAMMA', 'three', 'B', 'ALPHA', 'TWO', 'beta', 'z', 'FIVE', 'omega', 'z', 'SIX', 'omega', 'z')
('ONE', 'A', 'OMEGA', 'five', 'Z', 'OMEGA', 'four', 'C', 'GAMMA', 'three', 'B', 'ALPHA', 'TWO', 'beta', 'z', 'SIX', 'omega', 'z')
('ONE', 'A', 'OMEGA', 'six', 'Z', 'OMEGA', 'five', 'Z', 'OMEGA', 'four', 'C', 'GAMMA', 'three', 'B', 'ALPHA', 'TWO', 'beta', 'z')
('ONE', 'A', 'OMEGA', 'four', 'C', 'ALPHA', 'TWO', 'beta', 'b', 'THREE', 'gamma', 'z', 'FIVE', 'omega', 'z', 'SIX', 'omega', 'z')
('ONE', 'A', 'OMEGA', 'five', 'Z', 'OMEGA', 'four', 'C', 'ALPHA', 'TWO', 'beta', 'b', 'THREE', 'gamma', 'z', 'SIX', 'omega', 'z')
('ONE', 'A', 'OMEGA', 'six', 'Z', 'OMEGA', 'five', 'Z', 'OMEGA', 'four', 'C', 'ALPHA', 'TWO', 'beta', 'b', 'THREE', 'gamma', 'z')
Stage 2 has 6 outcomes

pairs_pool_5=[(['A', 'BETA'], ['B', 'ALPHA']), (['A', 'GAMMA'], ['C', 'ALPHA']),
               (['A', 'OMEGA'], ['Z', 'BETA']), (['A', 'OMEGA'], ['Z', 'GAMMA'])]
final_outcome=Counter()
for i in outcome_list_4[-1].keys():
    temp=run_probability_calculation_and_plot_IS110(i,pairs_pool_5,numberofevents=3,plot=0)
    final_outcome.update(temp[-1])
print(f'All reverts back to the original list {list(final_outcome.keys())}==([
    'ONE', 'A', 'ALPHA',
    'TWO', 'beta', 'b',
    'THREE', 'gamma', 'c',
    'FOUR', 'omega', 'z',
    'FIVE', 'omega', 'z',
    'SIX', 'omega', 'z'])')

[72] ✓ 0.0s Python

... All reverts back to the original list True
```