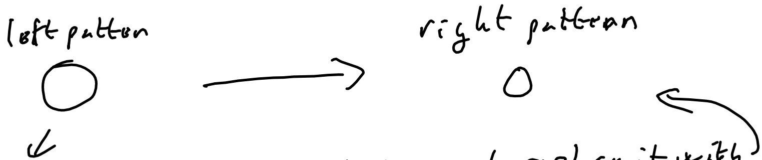
Kappa

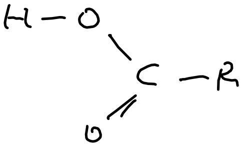
Juri Kolčák

Formal Languages

Formal languages are defined on strings, but the same **rule-based transformation** approach can be applied to different structures.

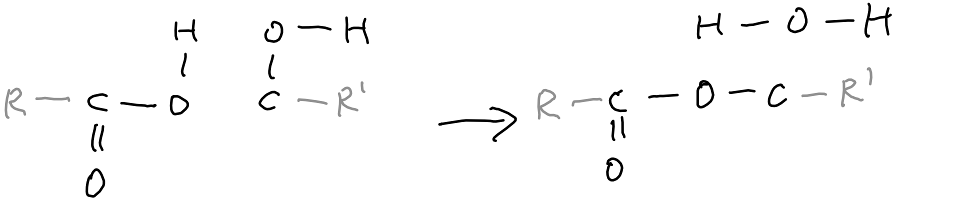


we find "match" in our state and replace it with



Carboxylic acid

Esturification: $RCOOH + R'COH \rightarrow RCOOCR' + H_2O$



Kappa

Kappa is a relatively young method, although graph transformation itself is much older.

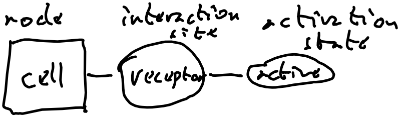
~1973

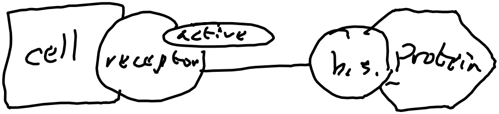
21st cen.
~2010

Applied to a special category of graphs, called **site graphs**.

site graph have nodes = agents/components of the system

each node has interaction sites





each interaction site can have an activation state
each interaction has a binding state

Each interaction site has to have a unique 'name'

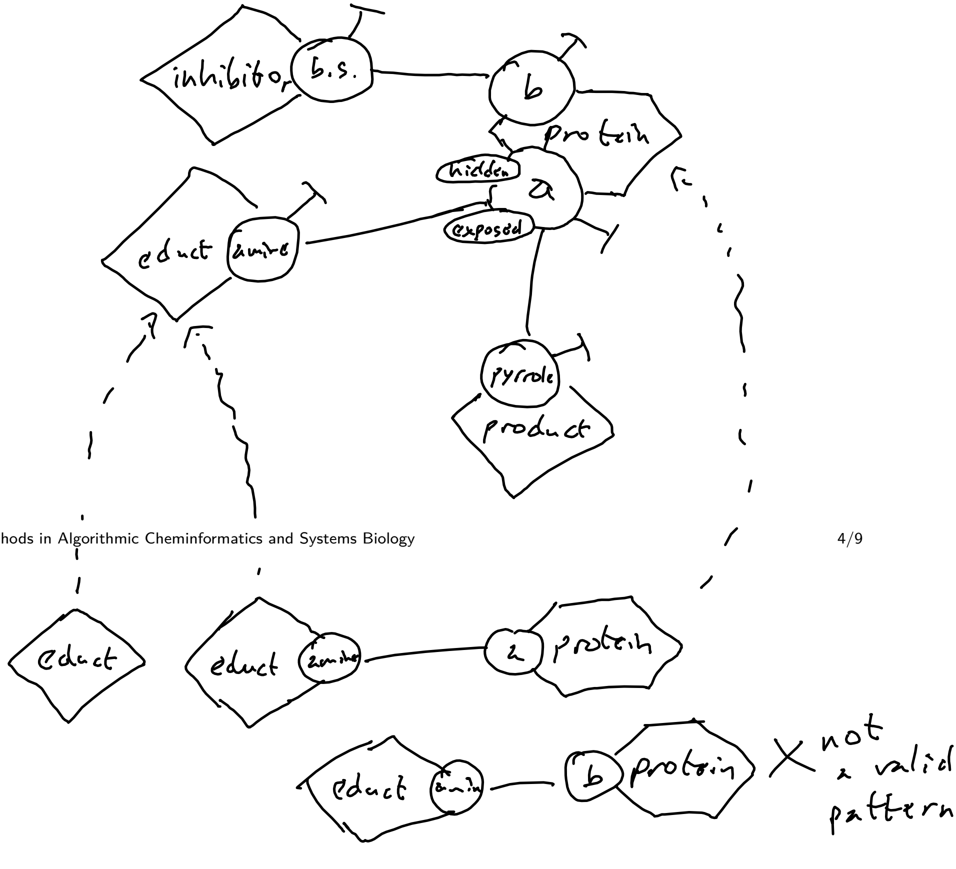
⇓

site graphs are rigid

Type Graph

signature / contact map

Effectively the "alphabet" of the model, specifying all types of nodes (agents), their interaction sites, activation and binding states.



Patterns

A pattern is a site graph for which the unique function mapping the pattern into the type graph is a homomorphism.

Patterns can also map into each other, these mappings are called **embeddings**.

The maximal, fully specified, patterns in which all nodes specify all the interaction sites given by the type graph, and each interaction site specifies a binding state as well as an activation state if the type graph specifies any activation states for said site, as called **complexes** and make up the states of the model.