




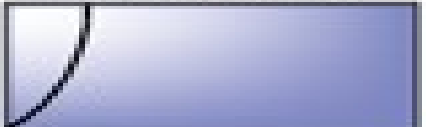

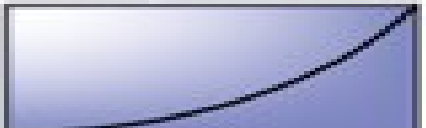


# ***Languages Of Genes***

Bairagi Nath Behera  
Nitish Rawat  
Sambhav Sharma

# Introduction

- In the same decade when DNA structure and genetic code for gene expressions were discovered Chomsky came out with his work on Formal Languages.
- Chomsky's work was based on the fact that we have infinite languages based on finite characteristics.
- As DNA and other biological sequences are richly expressive we can try to have a link between them and Formal languages.

# Chomsky Hierarchy

Language	Automaton	Grammar	Recognition
Recursively enumerable languages	<p>Turing machine</p> 	<p>Unrestricted</p> $Baa \rightarrow A$	<p>Undecidable</p> 
Context-sensitive languages	<p>Linear bounded</p> 	<p>Context sensitive</p> $At \rightarrow aA$	<p>Exponential?</p> 
Context-free languages	<p>Pushdown (stack)</p> 	<p>Context free</p> $S \rightarrow gSc$	<p>Polynomial</p> 
Regular languages	<p>Finite-state automaton</p> 	<p>Regular</p> $A \rightarrow cA$	<p>Linear</p> 

# Little More Terms

- Ambiguity- Two ways to reach a string
- Indexed Languages- Apart from Chomsky's basic languages, little modified one.
- Lindenmayer Systems- System for biological sequences based on IL
- Closure Properties and Decidability

# Nucleotide Linguistics

- **Structural Linguistics** :

Linguistic power required to encompass various phenomena observed in nucleic acids based on their structure.

- **Functional Linguistics** :

Linguistic power required based on the various information structure on basis of the different functions.

# Simple DNA helix

- $\Sigma_{\text{DNA}} = \{ g, c, a, t \}$
- Now, in our grammar we want the compliments of various base pair.
- Using simple regular expression we can inherit a string and then find it's complimentary strand.
- But this power isn't enough in cases we generally require in nucleiotides, RNA and proteins.

# Nucleic Acids Are not Regular

- Inverted repeats are prevalent features of nucleic acid.
- This implies that the substring and its reverse complement are both to be found on the same strand, which can thus fold back to base-pair with itself and form a stem-and-loop structure.
- Such base-pairing within the same strand is called secondary structure.
- Hence to do that we may be needing a CFL.

# CFL for Nucleotides

- $S \rightarrow bSb' \mid A$

$$A \rightarrow bA \mid \varepsilon$$

where  $b \in \Sigma_{DNA}$

- This structure may seem to be over as on with regular languages but no it fails as it requires a stack power because it's almost the same case as of we see in palindromes.



# Nucleic Acids aren't CFL

- As we saw in last way , it's for the structure that is Ideal.
- Ideal structure means having same number of complement strings present as of base.
- The illustration is that a string of direct repeats extending infinitely in either direction could shift an arbitrary number of times, and still maintain base-paired structure with its reverse complementary string through alternative “hybridization.”
- This make us move our grammar beyond CFL.

# Nucleic Acids as Indexed Languages

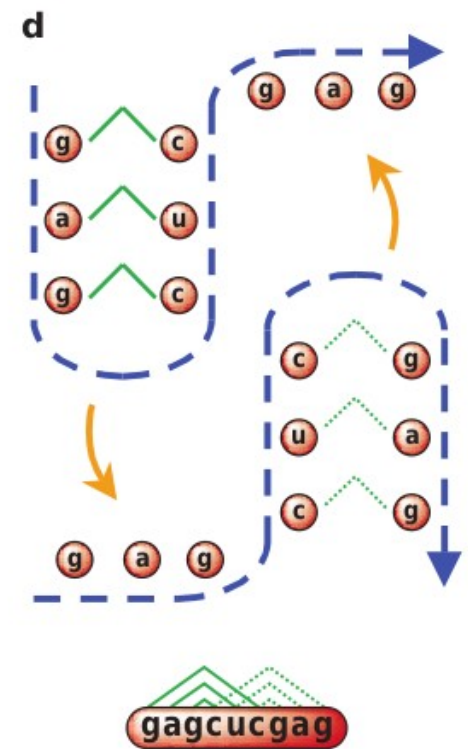
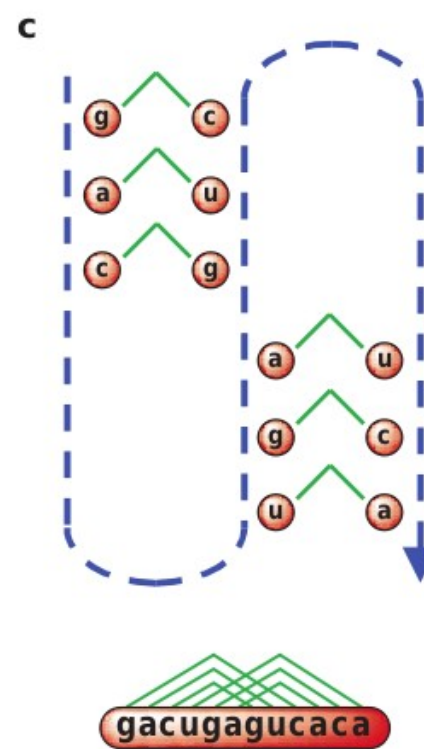
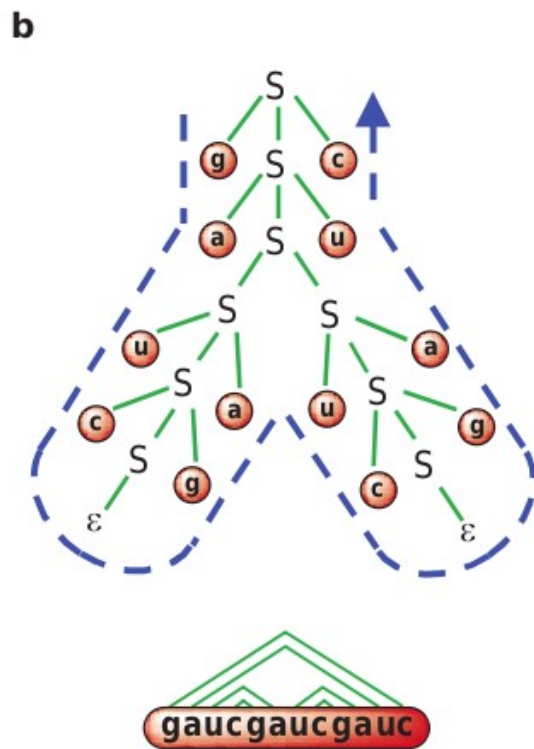
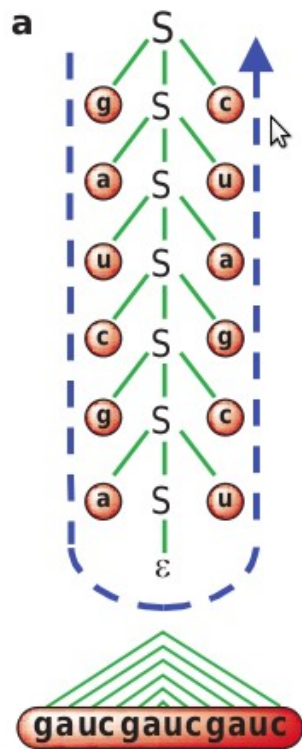
- The features described thus far are all encompassed by CSLs with  $\varepsilon$ , and in fact can be described by indexed grammars, which specify the IL subset of CSLs.

$$S \rightarrow bS^b \mid A$$

$$A^b \rightarrow Ab$$

$$A \rightarrow \varepsilon$$

# Grammar style derivations



So..

Using formalisms called tree-adjoining grammars which are considered to be mildly context-sensitive and relatively tractable, it is possible to encompass a wide range of RNA secondary structures.

Natural languages seem to be beyond context-free as well, based on linguistic phenomena entailing cross-serial dependencies.

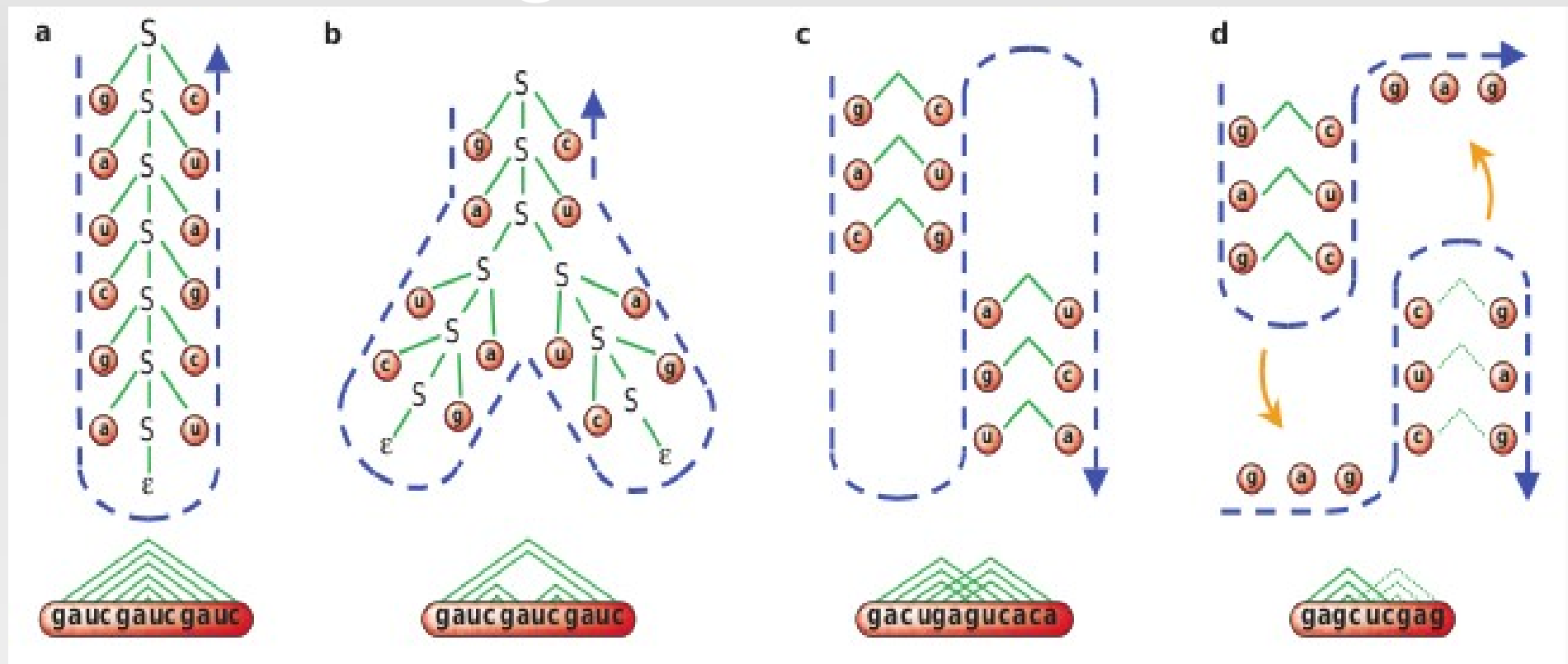
Thus by one measure nucleic acids may be said to be at about the same level of linguistic complexity as natural human languages.

# Protein Linguistic

There has been less activity in modelling proteins with linguistic methods, perhaps because they are viewed as having a richer basic interactions and conformations than nucleic acid.

Specific aspects of protein structure have been modelled explicitly with grammars. Secondary structural elements and in particular the hydrogen bonding between strands in a  $\beta$ -sheet may be arranged in antiparallel fashion or in parallel fashion. Such arrangements have been represented using stochastic tree grammar.

# Protein Linguistic



Grammar-style derivations of idealized versions of RNA structures. **a**, A stem; **b**, a branched structure; **c**, a pseudoknot; and **d**, alternative secondary structures of an attenuator. The trees for **a** and **b** are graphical depictions of derivations from grammars given in the text. By convention, a starting nonterminal **S** is at the root of the tree and gives rise to branches for each symbol to which it rewrites in the course of the derivation. The string derived can be read by tracing the frontier or leaf nodes of the tree, left to right (dashed blue lines). For **c** and **d**, derivation trees are not explicitly indicated because of the complexity of the context-sensitive grammars required<sup>7</sup>. The same strings are also shown in linear fashion, with dependencies indicated between terminals derived at the same steps.

# Protein Linguistic

Mathematicians are concerned with closure properties of language, that is, whether they remain at the same level of the Chomsky hierarchy.

Simple concatenation of strings is a so-called regular operation, whereas insertion of string in another is a context-free operation.

However, translocation of segments of a string may constitute an upward force in the Chomsky hierarchy that is inherent in evolution.

# Protein Linguistic

Greater richness of the language of genes and proteins indicates all the more the need for a well-founded descriptive paradigm.

This view will also allow us to expand our horizons beyond the relatively local phenomena of secondary structure, to large regions of the genome. This will allow us in turn to reason linguistically about processes of evolution.



# Computational linguistics and genes

The results summarized above all relate to structural aspects of macromolecules and independent of any information they contain.

Language process is often conceived as proceeding from (1) the lexical level, (2) the syntactical level (3) the semantic level, (4) the pragmatic level.

# Evolutionary Linguistics

- . Evolution
- . Application of Grammars for evolutionary linguistics.
- . in 1786, Darwins first proposed for evolutionary linguistics.
- .

# Repetition and Infinite languages

- Problem on RL
  - . if the string is greater then no.of states of FA
  - . if the string is less then no.of states of FA
- Problem on CFL
  - .unbounded duplications strings
- Move towards to CSL

# Mutation and Rearrangement

- change evolution—duplication, inversion, transposition, and deletion
- $\text{DUP}(L) = \{ xuuy \mid xuy \in L \}$
- $\text{INV}(L) = \{ xu^{-R}y \mid xuy \in L \}$
- $\text{XPOS}(L) = \{ xvuy \mid xuy \in L \}$
- $\text{DEL}(L) = \{ xy \mid xuy \in L \}$

# Mutation and Rearrangement cont..

- **definite clause grammar (DCG)**

- **Mutation :-**

point\_mutation([From],[To]).

(\_, point\_mutation(X,Y), \_) ==> Input/Output.

- **Rearrangements :-**

duplication, X, X --> X.

inversion, ~X --> X.

transposition, Y, X --> X, Y.

deletion --> X.

- **Evolution :-**

evolution --> [ ] | event, evolution.

event, X --> X,

(inversion | deletion | transposition | duplication)

# Inversion

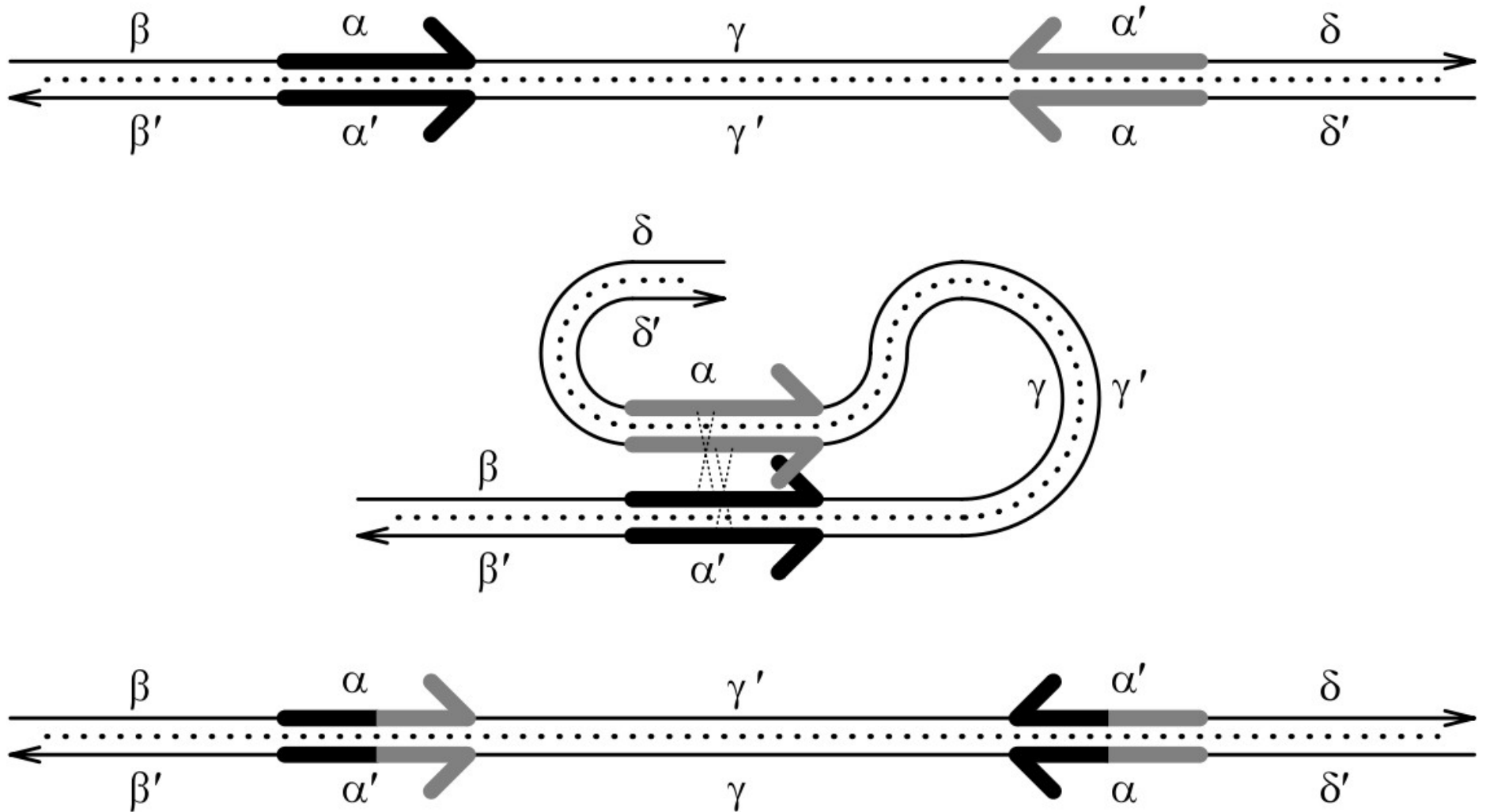


Figure 25. Inversion

# Excision and Integration

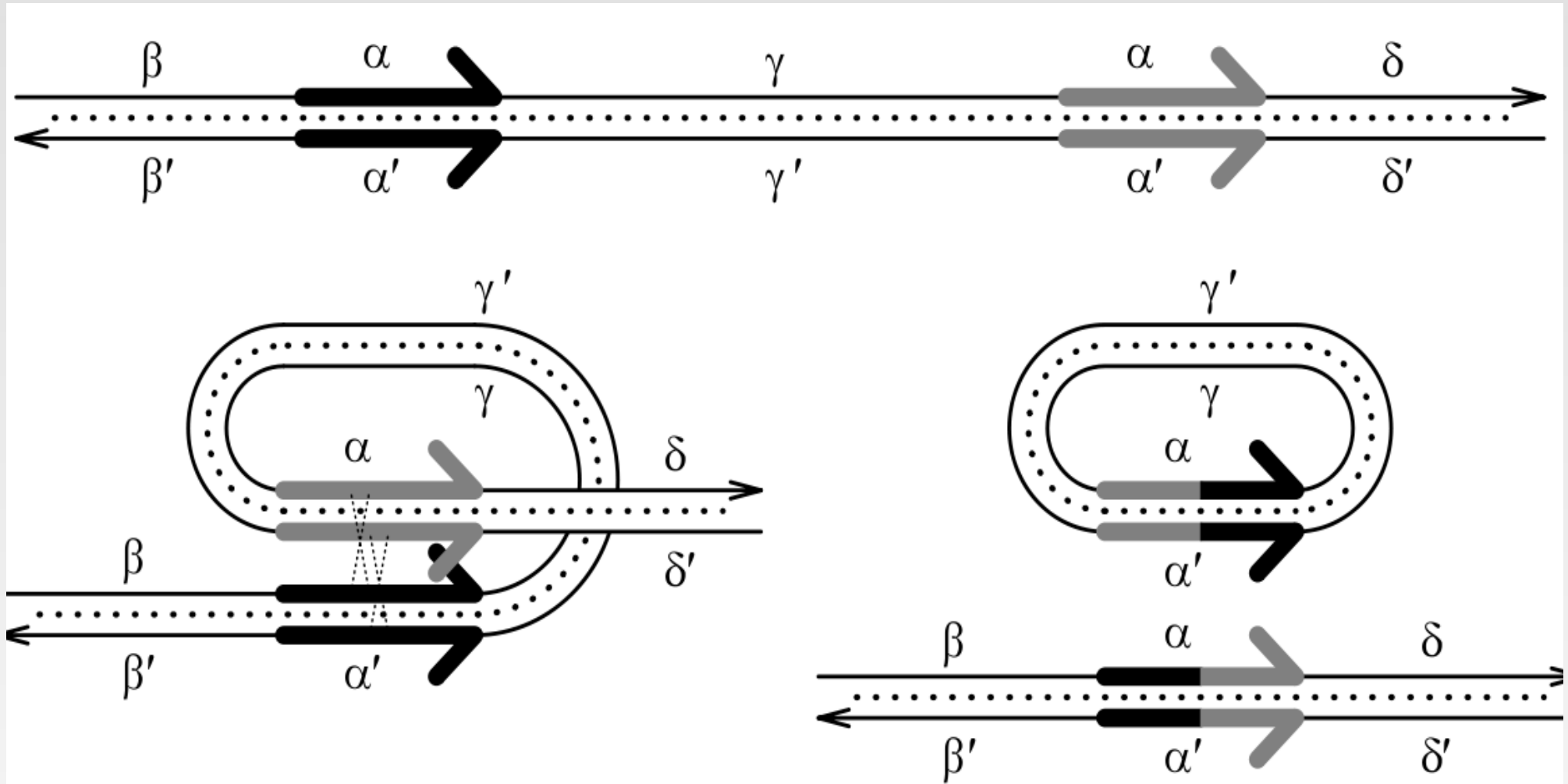


Figure 26. *Excision and Integration*

# DCG for Excision and Integration

- **Excision**

excision(O), S --> S, X, S,  
{(S,X)==>O/O}.

- **Integration**

integration(O), S, X, S --> S,  
{(\_,S,X,S,\_)==>O/O}.



# Comparison of Strings

- based on the detection of similarities between strings, rather than between a pattern and a string.
- “gaataattcggctta”\$Cost ==> “gacttattcgtagaa”
- We can implement this as a DCG:

```
[ ]$cost --> [ ].           % zero cost  
[H|T]$Cost --> [H],        % bases match; zero cost  
[H|T]$Cost --> [X],        % cost one  
[_|T]$Cost --> T$Ins,      % Cost is Ins+1  
String$Cost --> [], String$Del, % {Cost is Del+1}.
```

- Prolog “bagof” operation

```
best(From$Cost ==> To)  
bagof(Cost,(From$Cost ==> To),Bag),  
minimum(Bag,Cost).
```

# Phylogeny of Languages

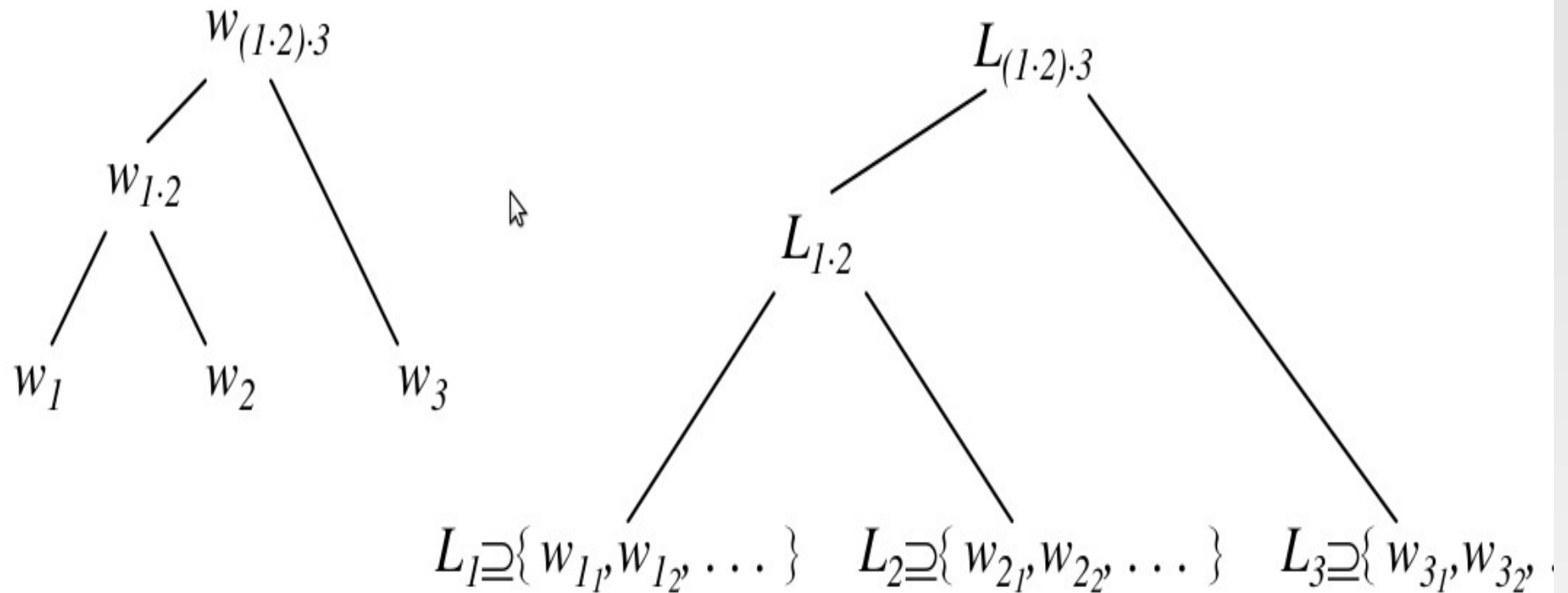


Figure 28. *Phylogenetic Trees of Strings (left) and Languages (right)*

# Conclusion

- Parse trees may reflect secondary structure
- Grammar nonterminals might model biochemical entities
- Grammar rules could describe intra-molecular interactions
- Greater-than-context-free grammars can model mutation and evolution
- Grammar derivation could model gene expression.
- Parsing might mimic certain biochemical processes

# References

- The language of genes - David B. Searls, Bioinformatics Division, Genetics Research, GlaxoSmithKline Pharmaceuticals.
- The Computational Linguistics of Biological Sequences - David B. Searls