Asilomar 2014

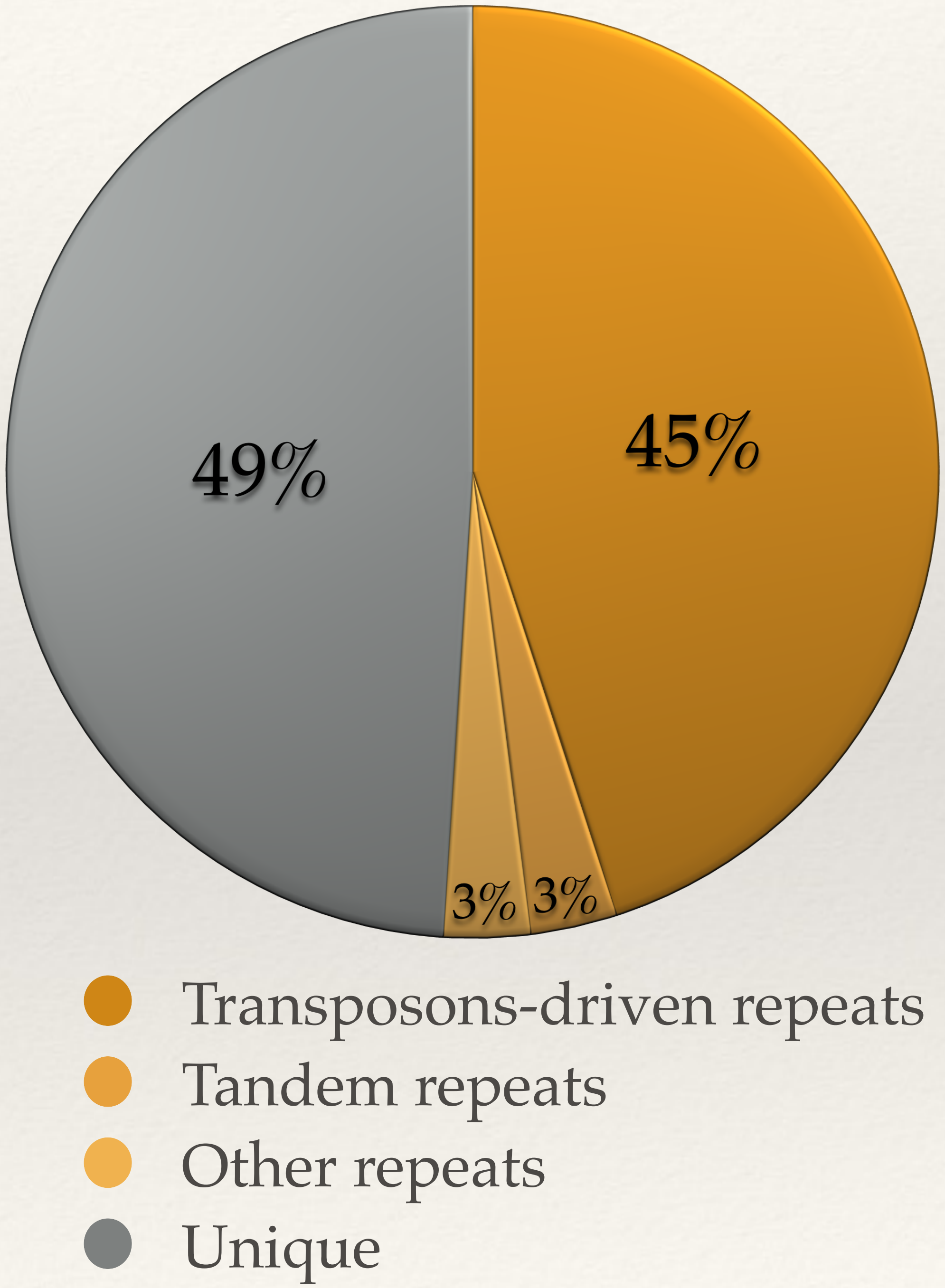
On the Capacity of String-Duplication Systems and Genomic Duplication

F. Farnoud🌴, M. Schwartz🔥, and J. Bruck🌴

🌴 California Institute of Technology
🔥 Ben-Gurion University of Negev

Repeated Sequences in Human Genome

- The majority of the human genome consists of repeated sequences.
  - Tandem repeats: TCATCATGCA
  - Transposon-driven repeats: TCATGCCATA
- Repeats provide a record of evolution and may cause chromosome fragility, expansion diseases, gene silencing, etc.



[Lander et al. Nature 2001]

Expressive Power of Repetitions

- "Much of the remaining 'unique' DNA must also be derived from ancient transposable element copies that have diverged too far to be recognized as such." [Lander et al. Nature 2001]

- Is it possible to generate a diverse family of sequences by duplication?

- Information theoretic view: *capacity* of *string duplication systems*.

A Tandem Duplication String System

- String system: starting string, (duplication) rule
- Example:
  - starting string = AGT
  - duplication rule: a substring of length 2 may be repeated in tandem

AGT

A Tandem Duplication String System

- String system: starting string, (duplication) rule
- Example:
  - starting string = AGT
  - duplication rule: a substring of length 2 may be repeated in tandem

$$AGT$$
$$\downarrow$$
$$AGAGT$$

A Tandem Duplication String System

- String system: starting string, (duplication) rule
- Example:
  - starting string = AGT
  - duplication rule: a substring of length 2 may be repeated in tandem

$$AGT$$

$$\downarrow$$

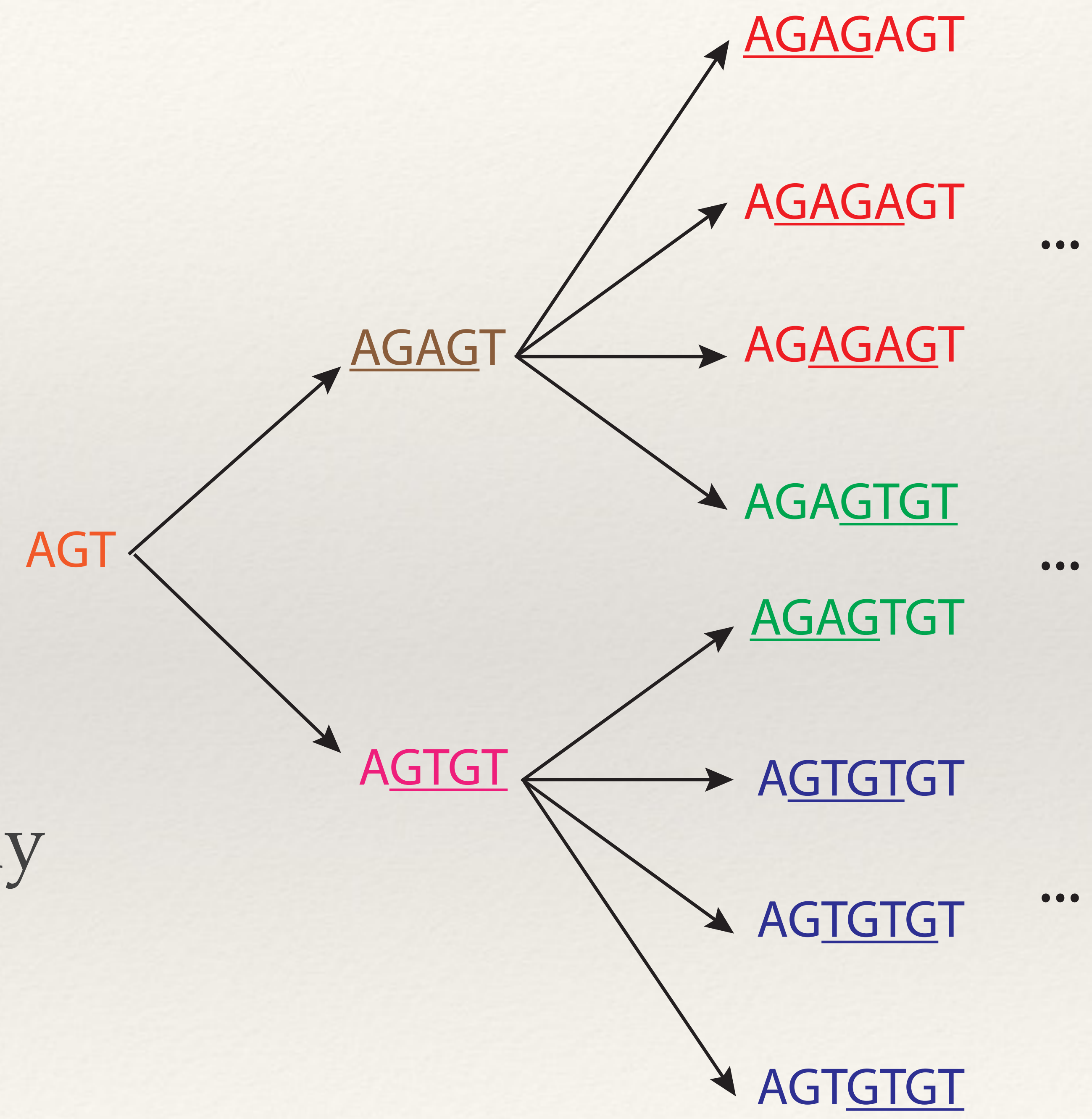
$$AGAGT$$

$$\downarrow$$

$$AGAGTGT$$

A Tandem Duplication String System

- String system: starting string, (duplication) rule
- Example:
  - starting string = AGT
  - duplication rule: a substring of length 2 may be repeated in tandem



String Duplication Systems

- *s*: starting string from an alphabet *A*
- *F*: family of duplication rules
- *S*=(*s*,*F*): sequences obtained by starting with *s* and applying functions $f \in F$.
- The *capacity* of *S* is given by

$$\mathsf{cap}(S) = \frac{1}{\log \delta(s)} \limsup_{n\to\infty} \frac{\log |S \cap A^n|}{n}$$

  - δ(s): # distinct symbols in *s*

Duplication Rules

Duplication Rules

- Duplication rules:

Duplication Rules

- Duplication rules:
  - **End duplication**: T<u>CAT</u>GC→ T<u>CAT</u>GC<u>CAT</u>

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC
  - **Reversed tandem duplication**: TCATGC→ TCATTACGC

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC
  - **Reversed tandem duplication**: TCATGC→ TCATTACGC
  - **Duplication with a gap**: TCATGC→ TCATGCATC

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC
  - **Reversed tandem duplication**: TCATGC→ TCATTACGC
  - **Duplication with a gap**: TCATGC→ TCATGCATC
- Parameters: length of duplicate k, gap k'

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC
  - **Reversed tandem duplication**: TCATGC→ TCATTACGC
  - **Duplication with a gap**: TCATGC→ TCATGCATC
- Parameters: length of duplicate k, gap k'
- Tandem duplication studied in literature: [Dassow'99,'02], [Leupold'04,'05]: Concerned with position in Chomsky hierarchy of formal languages.

Duplication Rules

- Duplication rules:
  - **End duplication**: TCATGC→ TCATGCCAT
  - **Tandem duplication**: TCATGC→ TCATCATGC
  - **Reversed tandem duplication**: TCATGC→ TCATTACGC
  - **Duplication with a gap**: TCATGC→ TCATGCATC
- Parameters: length of duplicate k, gap k'
- Tandem duplication studied in literature: [Dassow'99,'02], [Leupold'04,'05]: Concerned with position in Chomsky hierarchy of formal languages.
- Study of these fundamental systems is a step towards modeling complex biological systems.

End Duplication

- $F_{k,\mathrm{end}}$: set of functions duplicating a k-substring and appending it to the end
  - T<u>CAT</u>GC→ T<u>CAT</u>GC<u>CAT</u> (k=3)

End Duplication

- $F_{k,\text{end}}$: set of functions duplicating a k-substring and appending it to the end
  - TCATGC→ TCATGCCAT   (k=3)

Theorem: For any positive integer k, and S=(s,$F_{k,\text{end}}$), we have cap(S)=1.

End Duplication: Proof

Theorem: For any positive integer k, and $S=(s,F_{k,\text{end}})$, cap(S)=1.

- If k=1, every symbol can be appended to the end $\Rightarrow$ cap(S) = 1
  - eg. $s=ACG \rightarrow ACGA \rightarrow ACGAG \rightarrow ACGAGG \rightarrow \ldots$
- Proof outline:
  - Generate a string containing all possible k-substrings
    - $s=ACG$, k=2:

$$ACG\ AC\ G\underline{A\ A}C\ G\underline{A\ C}G\ G\underline{A\ G}A\ A\underline{C\ A}C\ A\underline{C\ C}G\ldots$$

  - Now in each duplication step, any k-substring can be duplicated.

Tandem Duplication

- $F_{k,\text{tan}}$: set of functions duplicating a k-substring and insert the duplicate immediately after original copy.
  - T<u>CAT</u>GC→ T<u>CATCAT</u>GC (k=3)
- Capacity in complete contrast to end-duplication

Tandem Duplication

- $F_{k,\tan}$: set of functions duplicating a k-substring and insert the duplicate immediately after original copy.
  - TCATGC→ TCATCATGC (k=3)
- Capacity in complete contrast to end-duplication

Theorem: For any positive integer k, and $S=(s,F_{k,\tan})$, we have

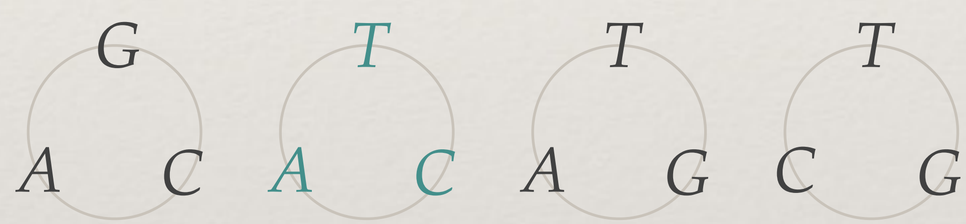
$$\text{cap}(S)=0.$$

Tandem Duplication: Proof

Theorem: For any positive integer k, and $S=(s,F_{k,\tan})$, $\mathrm{cap}(S)=0$.

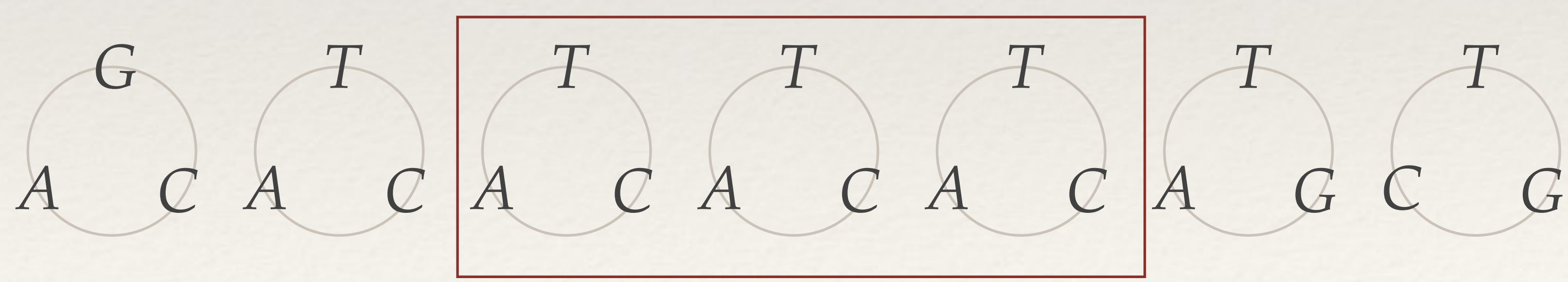
- Proof outline for $s=GCATGC$ and $k=3$
  - Map strings to sequence of circular k-substring:

$GCATGC \longrightarrow$



  - Duplication becomes repetitions of circular elements:

$GC\underline{AT\color{darkred}CAT}GC \longrightarrow$



  - Polynomial growth.

Tandem Duplication with Variable Length

- $F_{\geq k,\tan} = \{F_{i,\tan} : i \geq k\}$.
  - TCATGC→ TCATCATGC→ TCATCTCATGC (k=2)

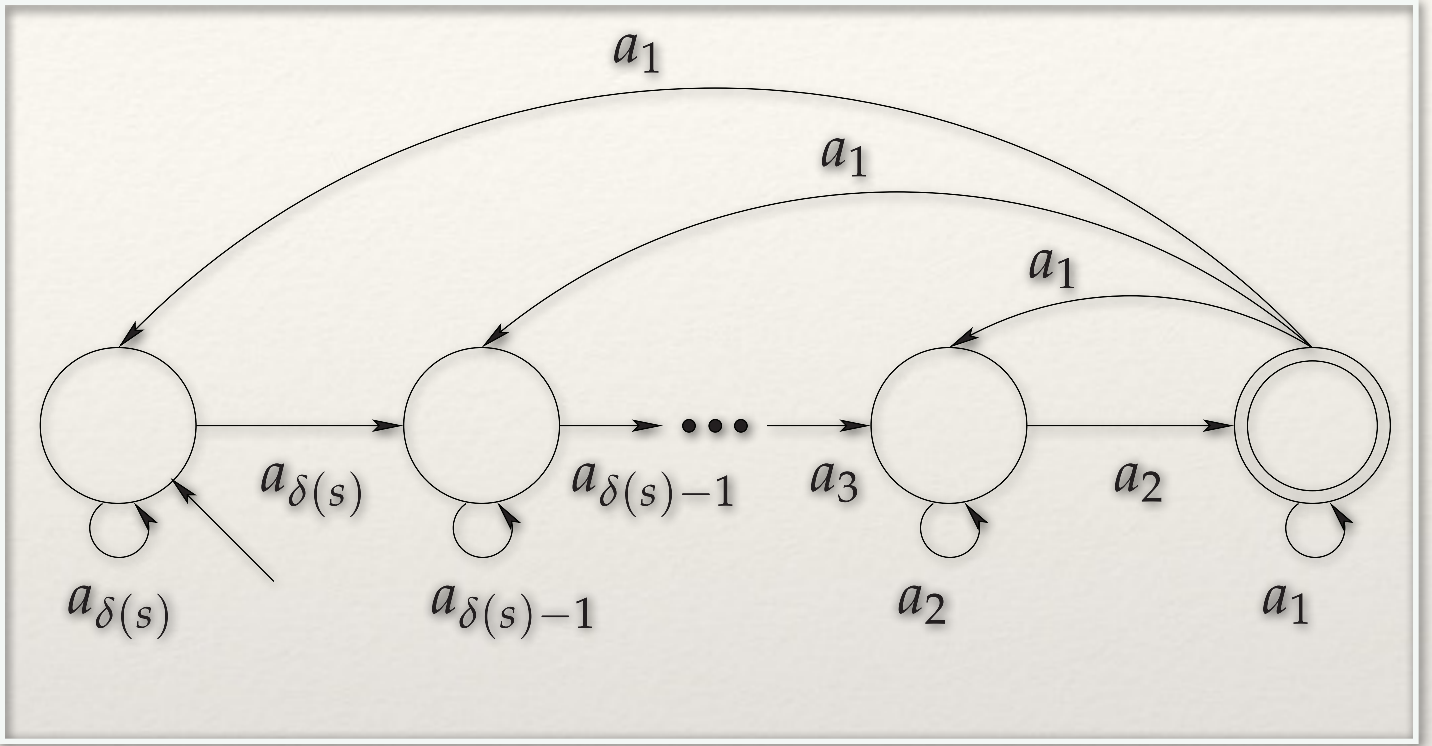
Theorem: For a nontrivial string s, let S=(s,$F_{\geq k,\tan}$) and S'=(s,$F_{\geq 1,\tan}$). We have cap(S)>0 and cap(S')≥$\log_2(r+1)/\log_2\delta(s)$, where r is the largest (real) root of

$$x^{\delta(s)} - \sum_{i=0}^{\delta(s)-2} x^i$$

$\delta(s)$	2	3	4	5
cap(S')≥	1	0.77	0.65	0.58

Tandem Duplication with Variable Length

- Outline of proof:
  - S' has a regular sub-language.
  - Capacity of sub-language is a lower bound.
  - Number length n words in sub-language equals number of length n paths in the automaton.
  - Capacity of sub-language is largest eigenvalue of adjacency matrix of automaton.



Finite automaton representing the regular sub-language.

Reverse Tandem Duplication

- $F_{k,\mathrm{rt}}$: set of functions duplicating a k-substring and inserting it immediately after original copy in *reverse*.
  - T<u>CAT</u>GC→ T<u>CATTAC</u>GC (k=3)
- Reversing the copy is seemingly a small change, but leads to nonzero capacity:

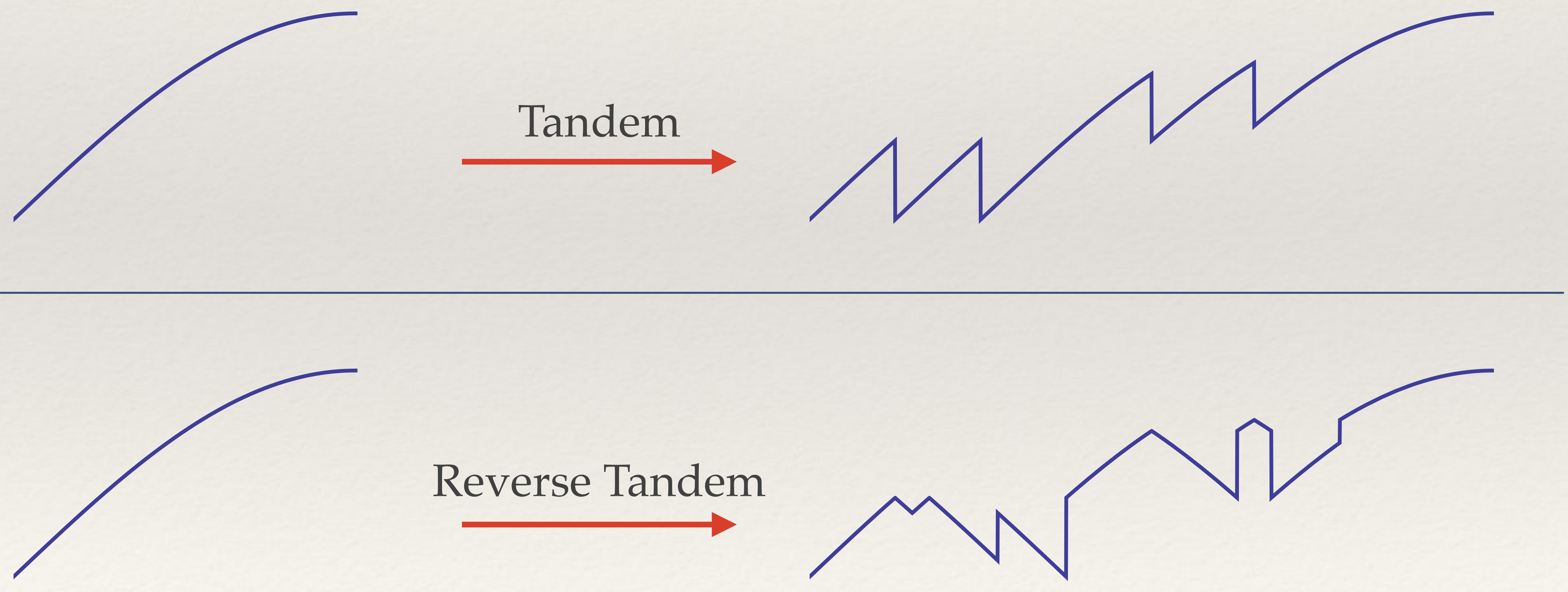
Theorem: For any positive integer $k>1$, and $S=(s,F_{k,\mathrm{rt}})$, we have

$$\mathrm{cap}(S)>0,$$

unless $\delta(s)=1$.
Furthermore, capacity depends on s, only through $\delta(s)$.

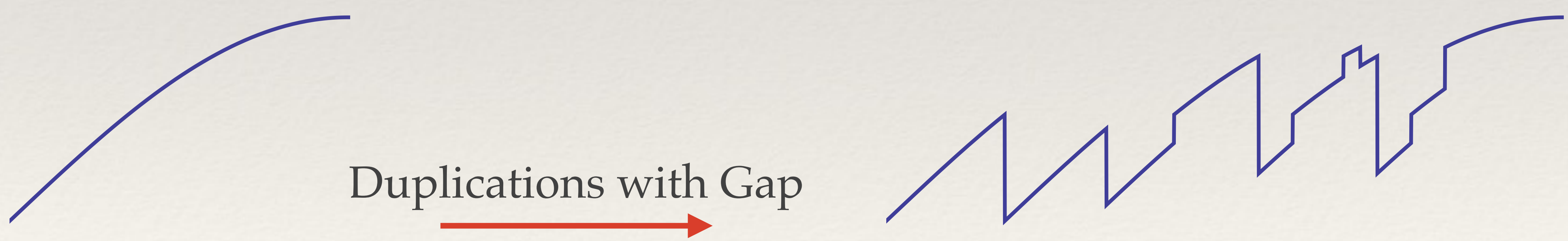
(Reverse) Tandem Duplication

- The main difference between tandem and reverse tandem duplication is that the former leads to near-periodic behavior with period *k*, but the latter does not.



Duplication with Gap

- $F_{k,k',\mathrm{gap}}$: set of functions duplicating a k-substring and inserting it k positions after original copy.
  - TCATGC→ TCATGCATC (k=3,k'=1)



Duplication with Gap

Theorem: The capacity of S=(s, $F_{k,k',\text{gap}}$) is zero if and only if s is periodic with period gcd(k,k').

- "if" direction:
  - If s is periodic with period gcd(k,k'), then so is every other string in S:
    - $k=2$, $k'=4$, $s=AGAGAGAG \Rightarrow S=\{(AG)^m: m\geq 4\}$

Duplication with Gap

Theorem: There are strings s such that for S=(s, $F_{k,k',\text{gap}}$) we have 0<cap(S)<1.

Theorem: If gcd(k,k')=1, then the capacity of S=(s, $F_{k,k',\text{gap}}$) depends on s only through $\delta(s)$.

Summary of Results

	0	$0 < \text{cap}(S) < 1$	1
End Duplication	✗	✗	✓
Tandem	✓	✗	✗
Tandem $\geq k$	✗	?	?
Reverse Tandem	✗	?	?
Gap (k,k')	✓	✓	?

Conclusion

- Studied expressive power of languages generated by different duplication rules from an information theoretic point of view.
- Except in very restrictive cases, duplication systems have nonzero capacity.
- These results *suggest* that it is plausible to generate diverse genomic sequences using duplications.