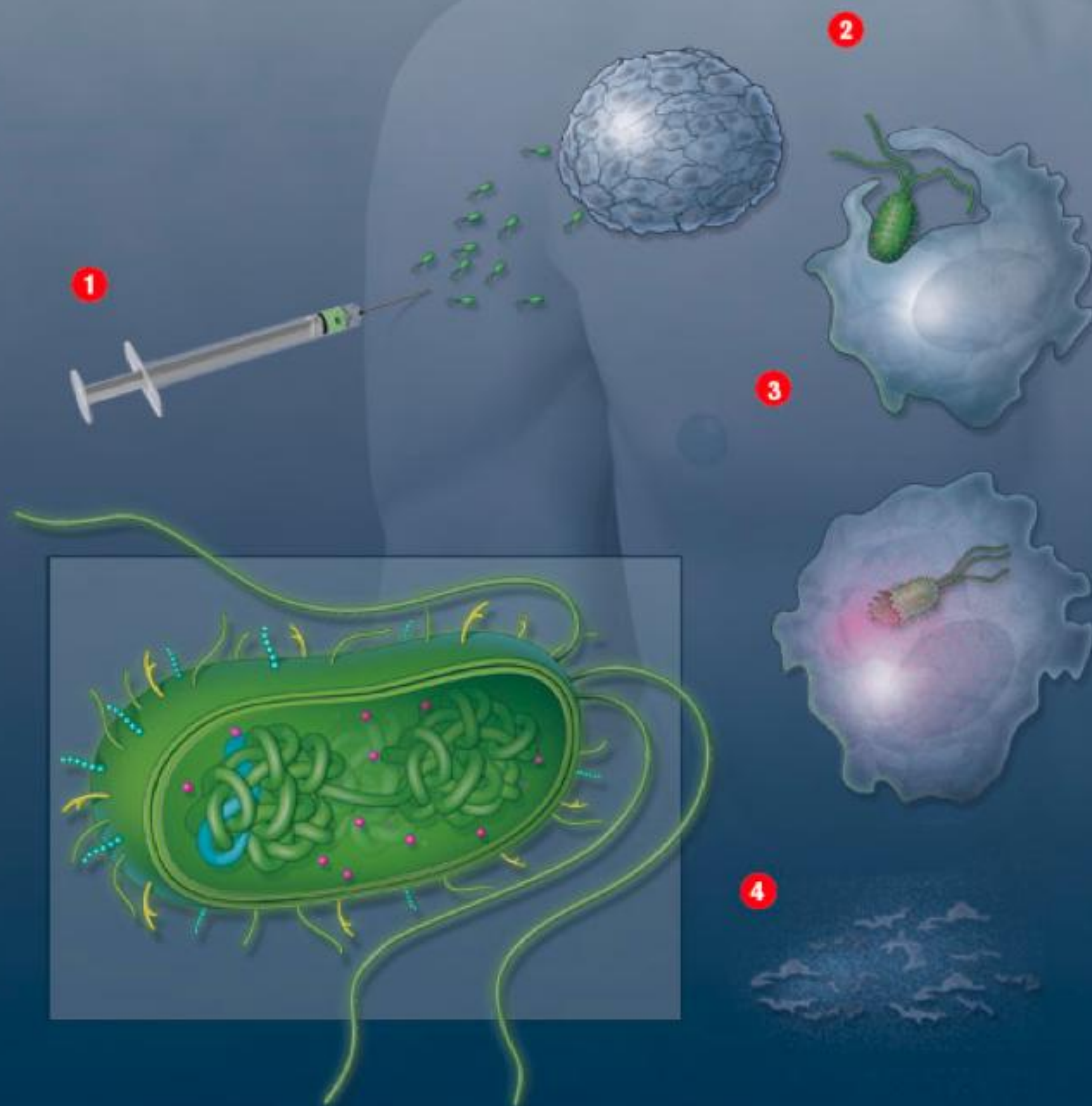


Towards Scalable Synthetic Biology

http://genomics.lbl.gov

arkin laboratory

The Enormous Potential of Engineered Microbes

Chemical Factories

Beyond the Bioreactor

Drugs

Bioremediation

Food Additives (Carotenoids)

Anticancer Therapeutics

Environmental Sensors

Chemical Feedstocks

Fuels

Adapted from Arkin Nature Biotech, 2008

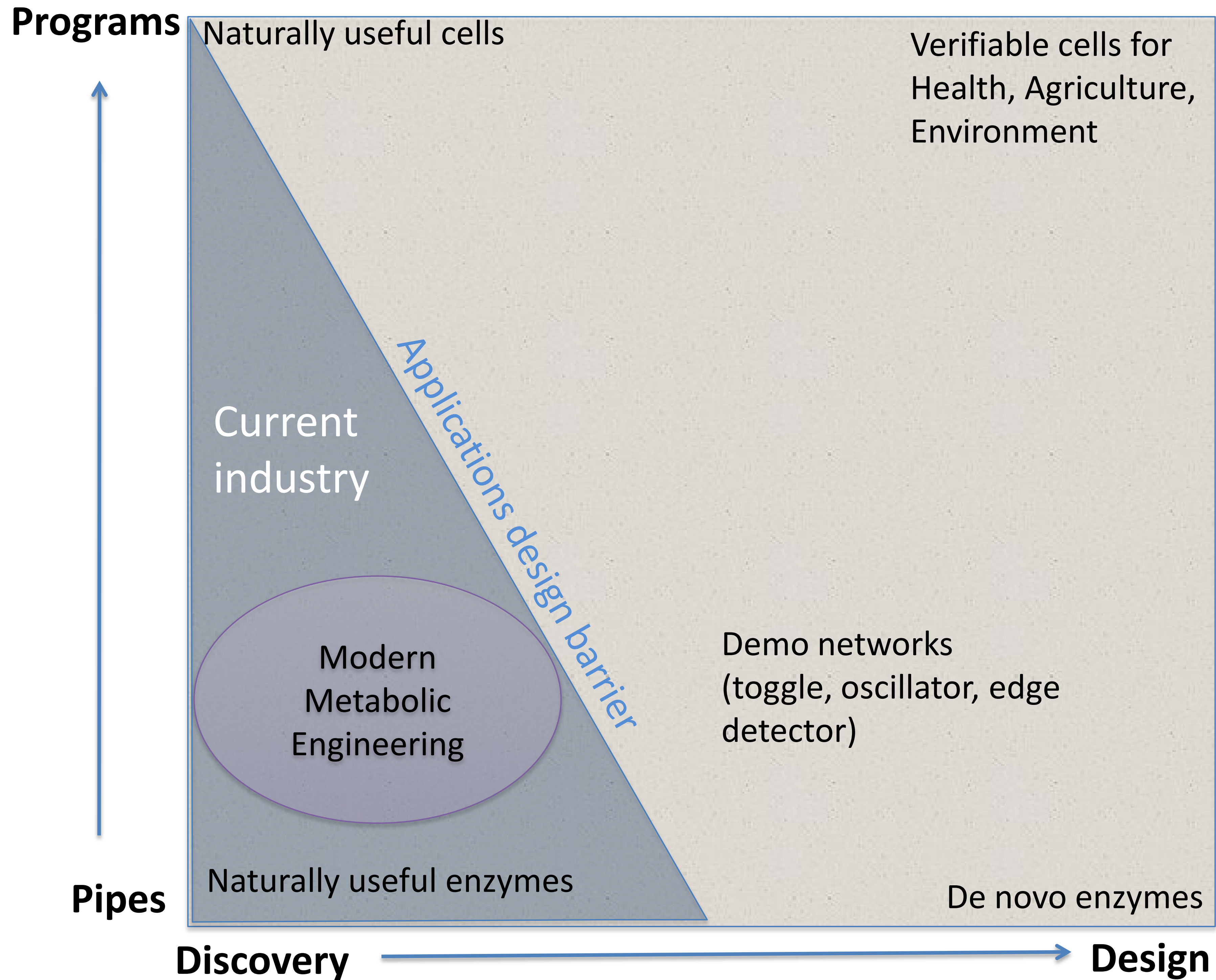
1) Engineered microbial solutions are being developed for energy, ecology and medicine.

2) The core of these solutions is gene expression.

Skerker, **Lucks**, Arkin Genome Biol. 2009

Verified Biological Design (VerBiD)

arkin laboratory
of systems and synthetic biology



arkin laboratory
of systems and synthetic biology

Scalable Genetic Engineering

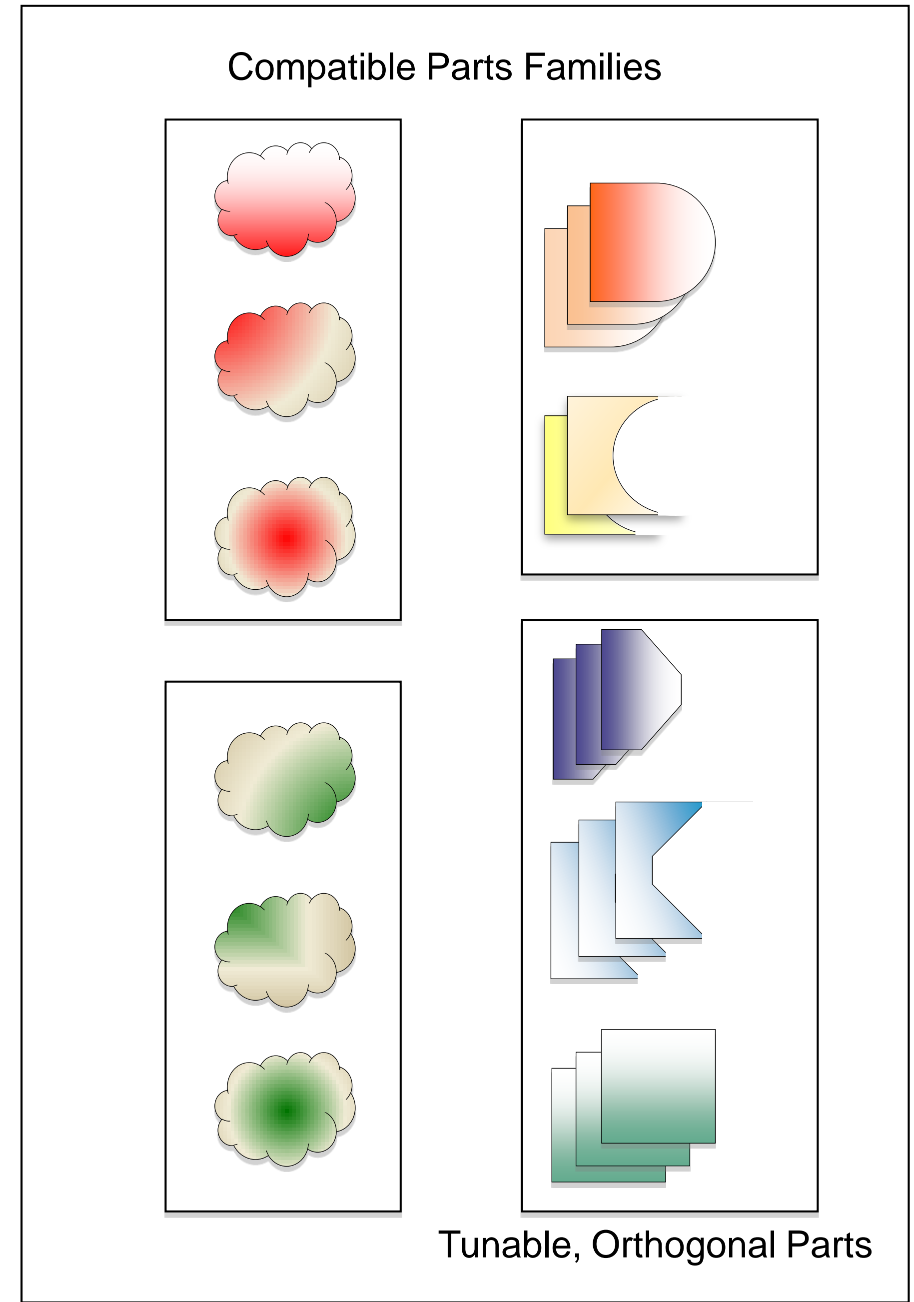
Cellular Environment

Cellular Resource

Composability

Composability

Independent Subsystems

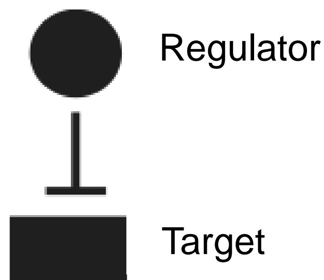


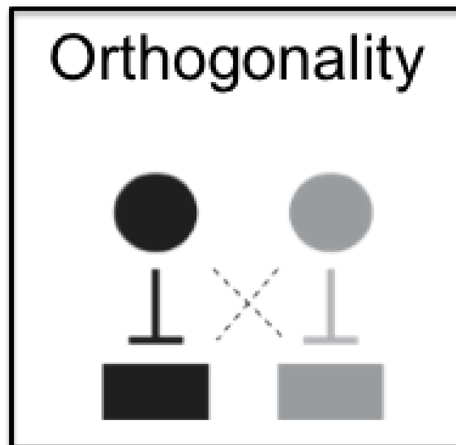
Goal: Develop Building Blocks that are Tunable, Orthogonal, Composable and Physically Homogeneous

Lucks, Qi, Whitaker, Arkin Curr. Op. Microbiol. 11, 2008

Designing Gene Regulation - Concepts

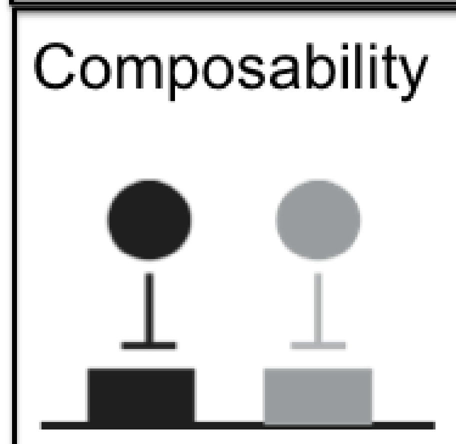
arkin laboratory
of systems and synthetic biology





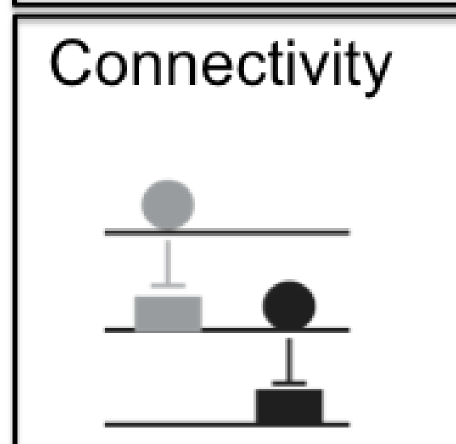
Concept: Regulators do not interfere with each other. (Specificity, Cross Talk)

Why need: Every regulator must be independent of the others.



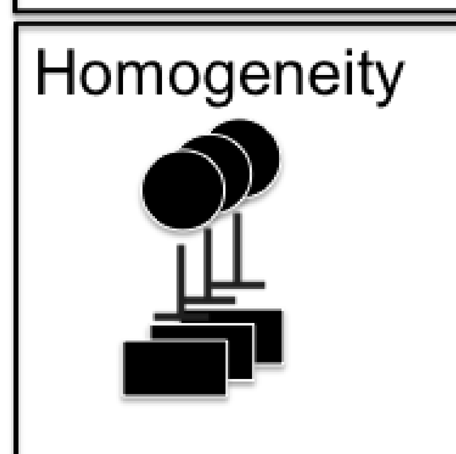
Concept: Regulators can be fused to give composite function. (Signal Integration)

Why need: A route to engineering more sophisticated regulation.



Concept: Regulators can be chained together. (Cascades, Feedback)

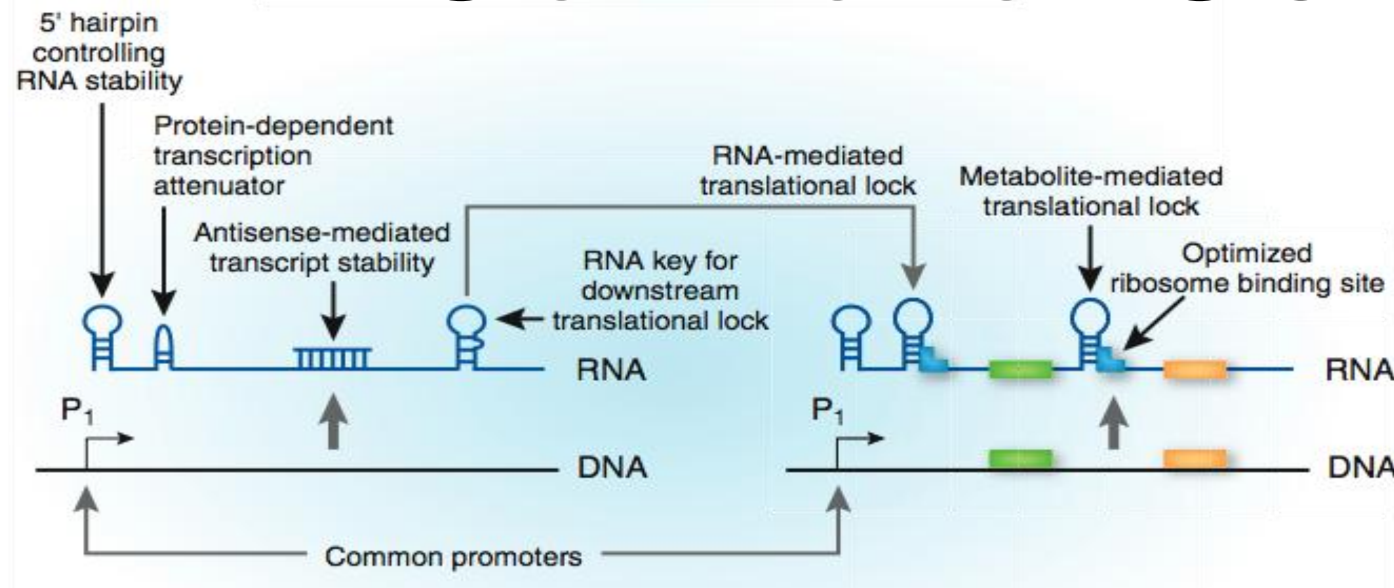
Why need: Necessary for signal propagation and feedback.



Concept: Regulators should obey very similar physics

Why need: Necessary for predictability/efficiency.

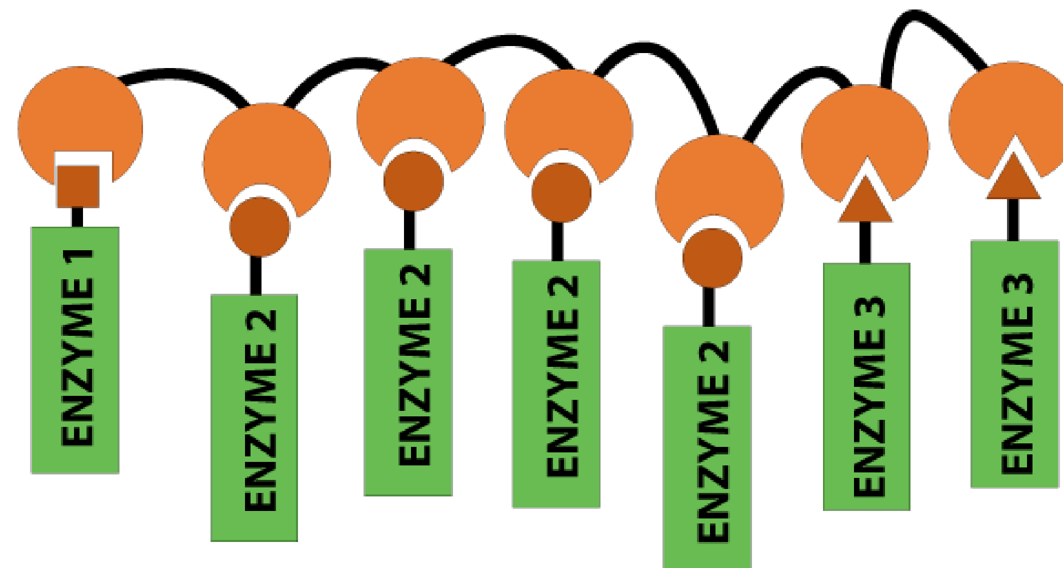
Control of Central Dogma



Nat Biotech, 2008

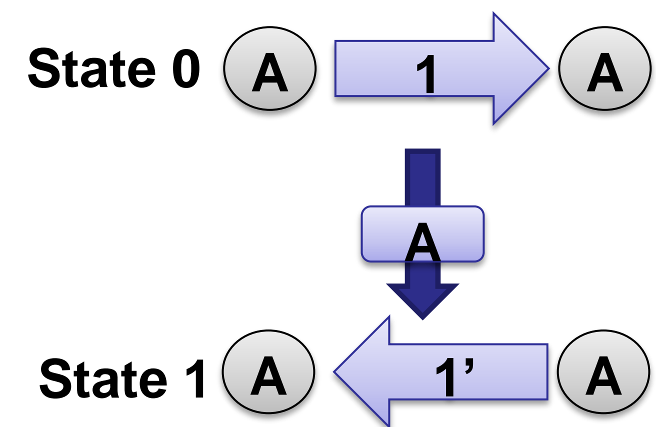
Julius Lucks
Lei Stanley Qi
Vivek Mutalik

Control of Protein-Protein Interaction and Localization



John Deuber
Weston Whitaker
Jay Keasling

DNA Writing

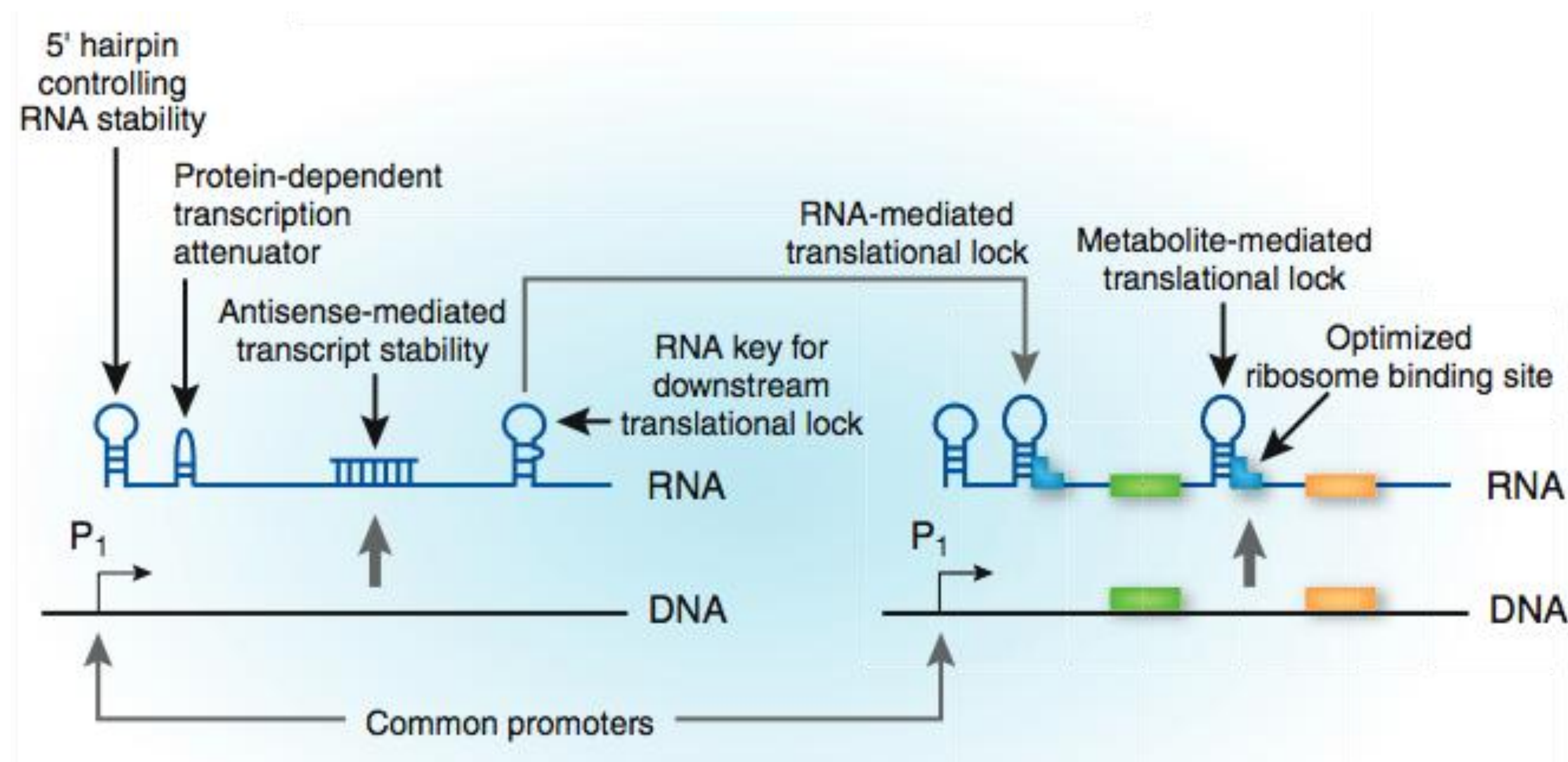


Michael Samoilov
Tim Ham

Gene Expression Engineering

$$\frac{dmRNA}{dt} = <OC><1-TAF> - <Deg_m> * mRNA$$

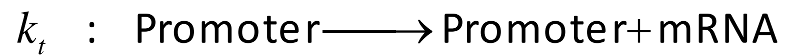
$$\frac{dP}{dt} = <Trans> * mRNA - <Deg_p> * P$$

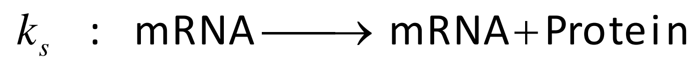


Nat Biotech, 2008

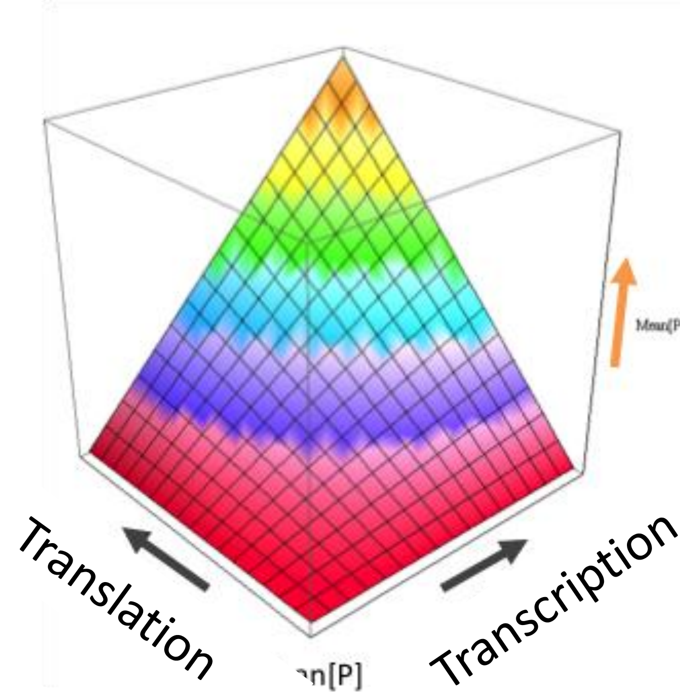
Quality

Vastly simplified model

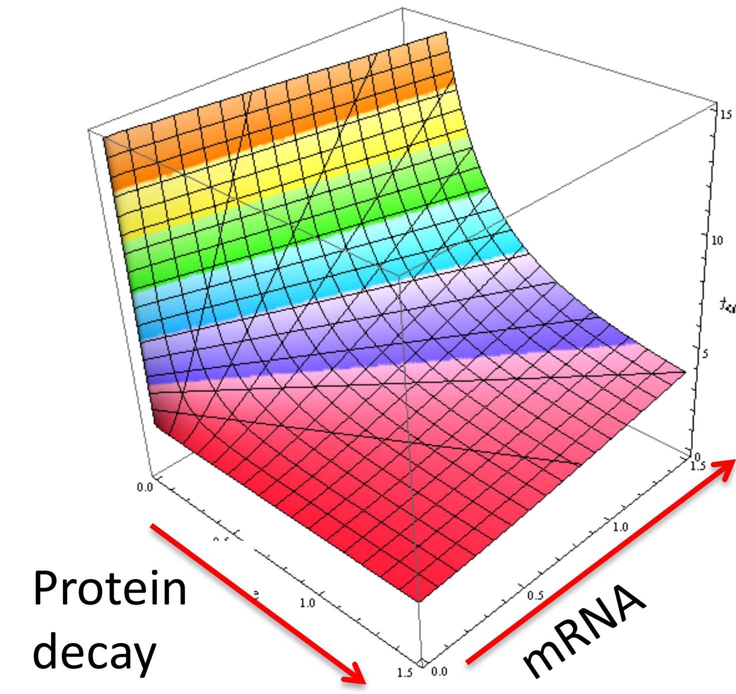




Mean protein levels

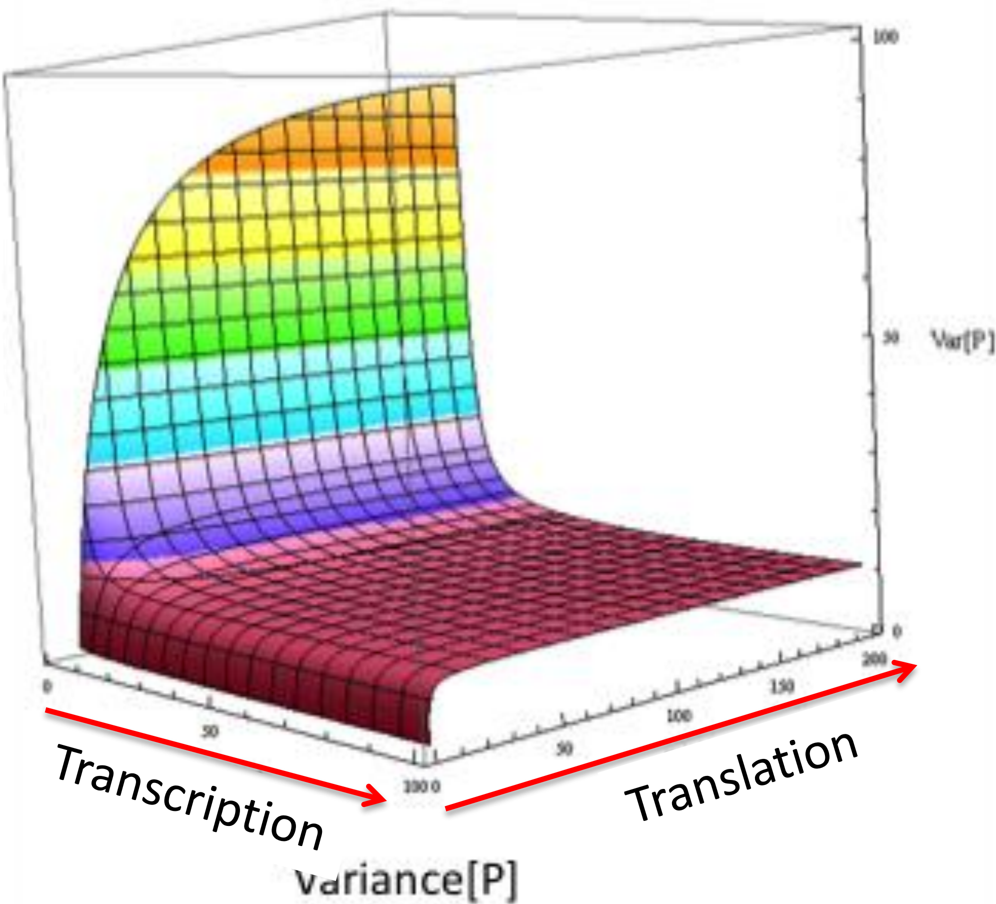


Time to 90% Mean



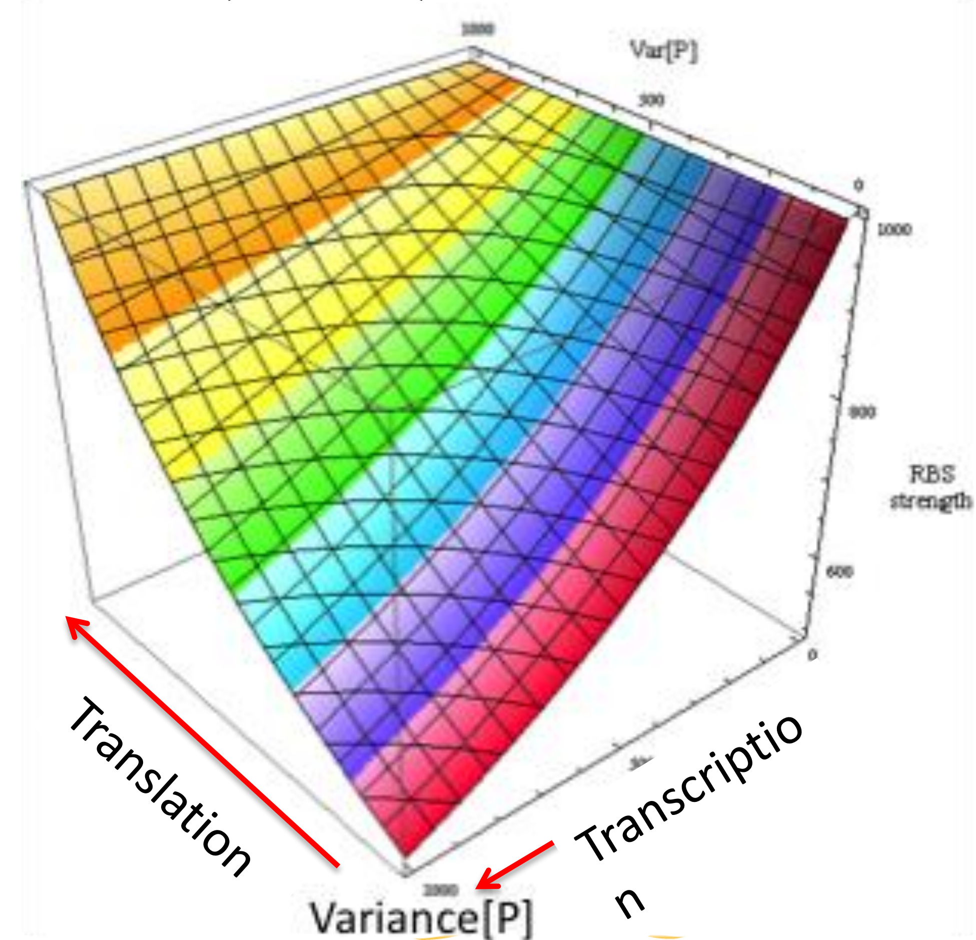
Variance of protein levels

$\langle \text{Protein} \rangle = Const \lesssim 100$

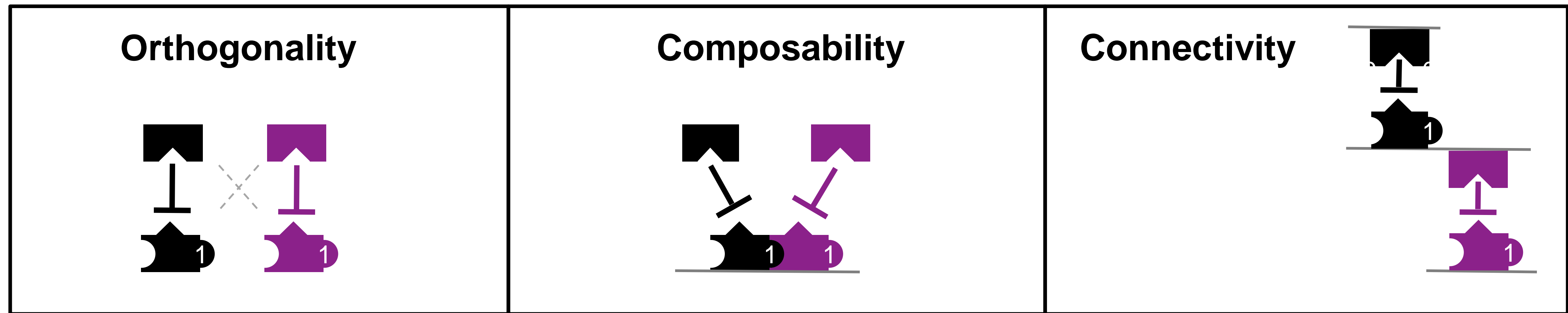


Same mean,
Different
variances

$\langle \text{Protein} \rangle = Const > 100$



Three Stories



Many regulators to choose from

DNA

Regulatory | Coding Sequence | Regulatory

Transcription

RNA

Regulatory | Coding Region | Regulatory

Translation

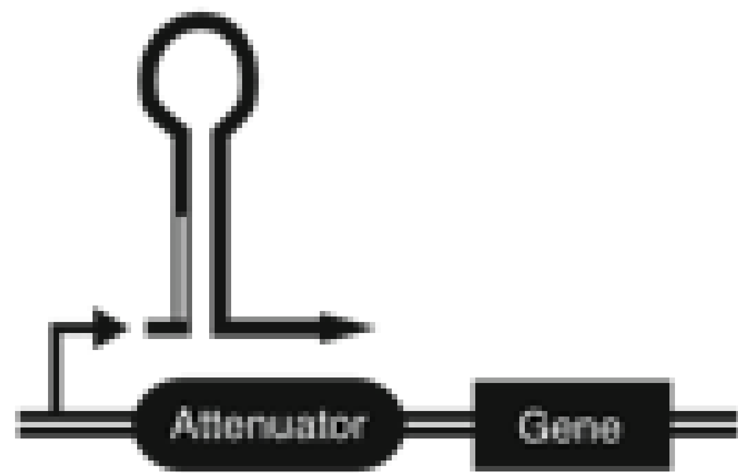
Protein

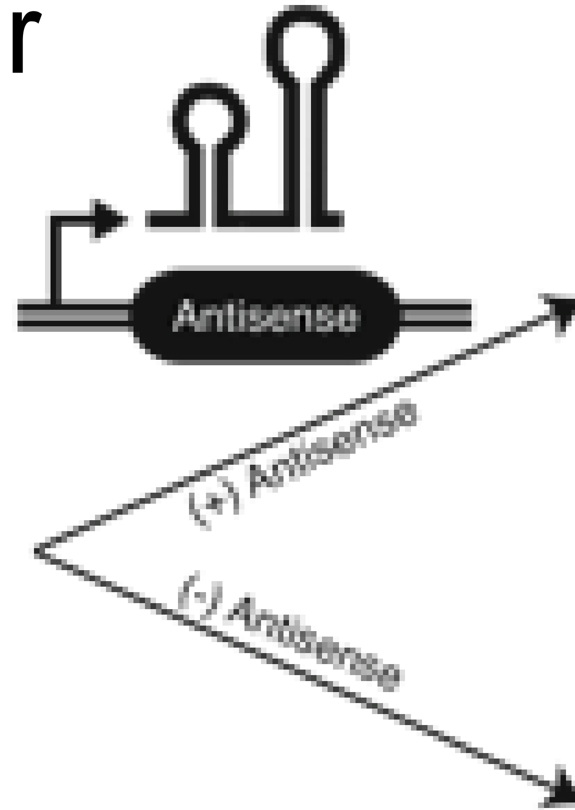
P

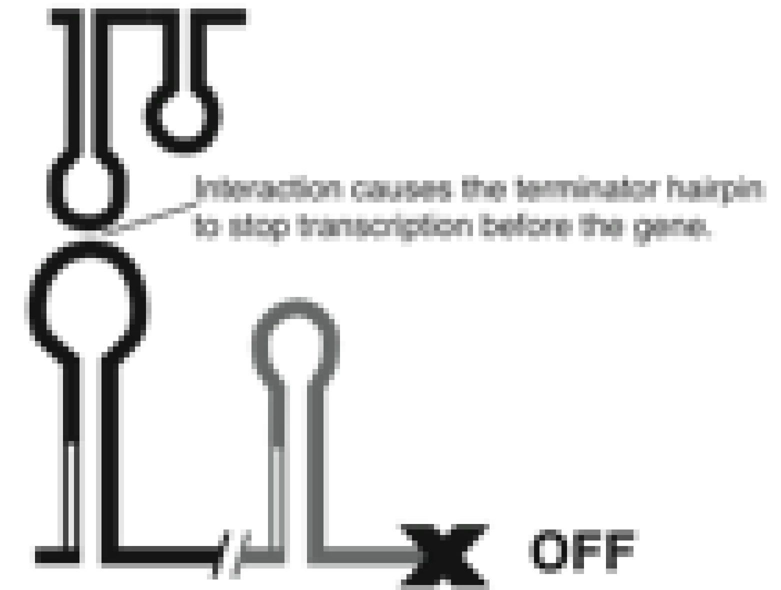
Mechanism	Molecule (RNA ~, Protein ●)
Transcription Factors	●
Transcription Attenuators	~ ●
Riboswitches	
RNA Binding Proteins	~ ●
Translational Repressors	
Self-Splicing Introns	
Transcription Terminators	~
Self-Splicing Ribozymes	
Post-Translational Modification	●

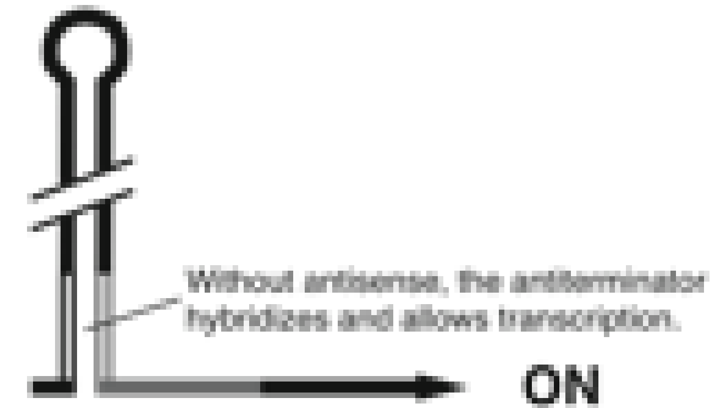
PT181 Attenuator

A

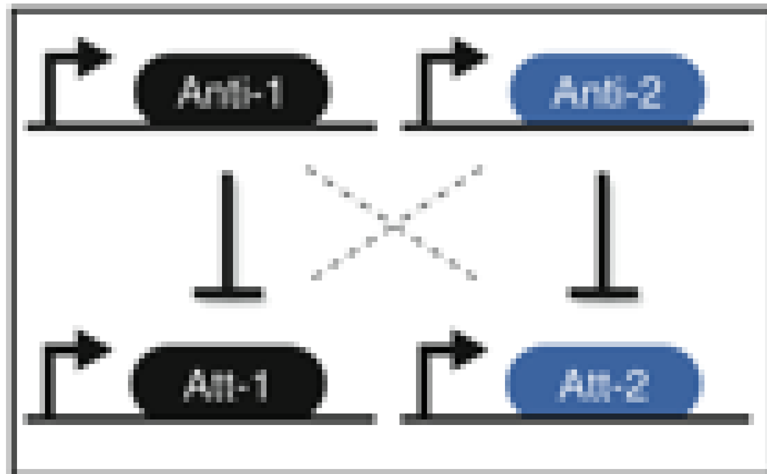


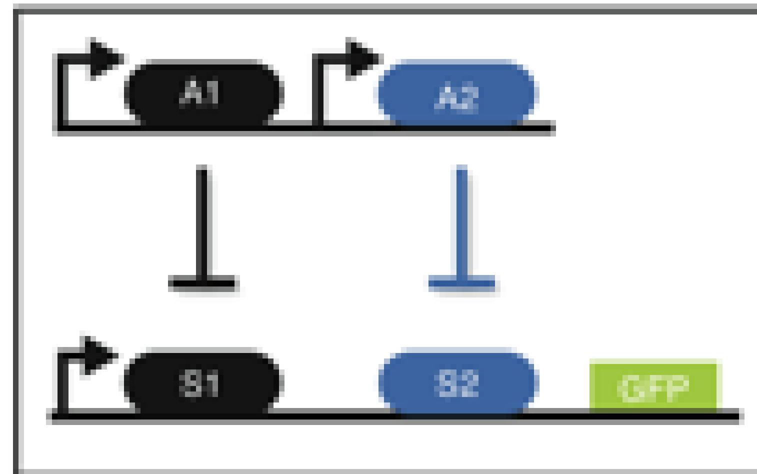


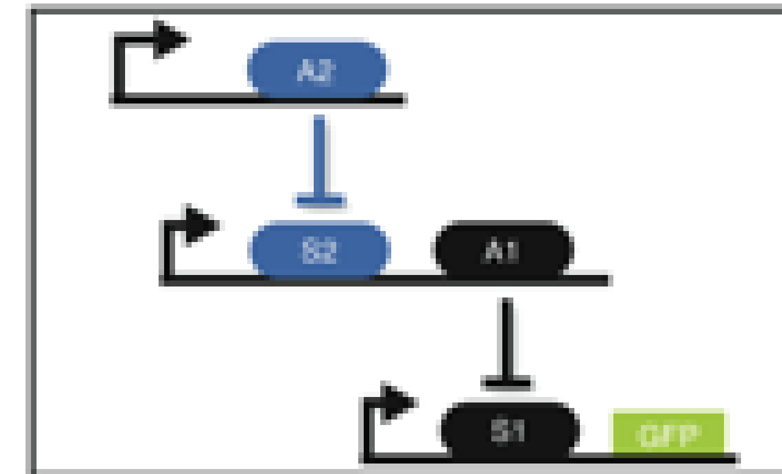




B

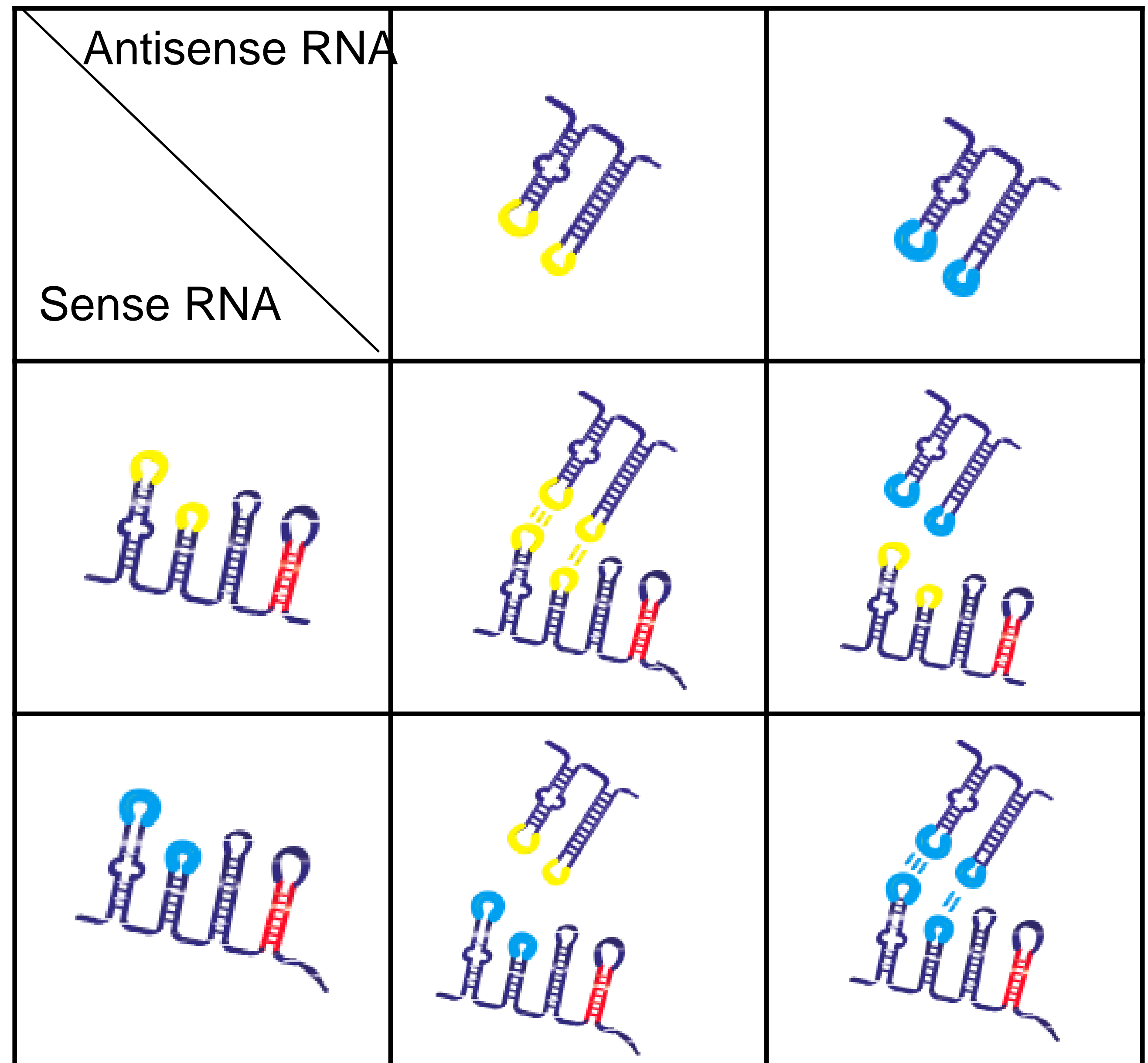




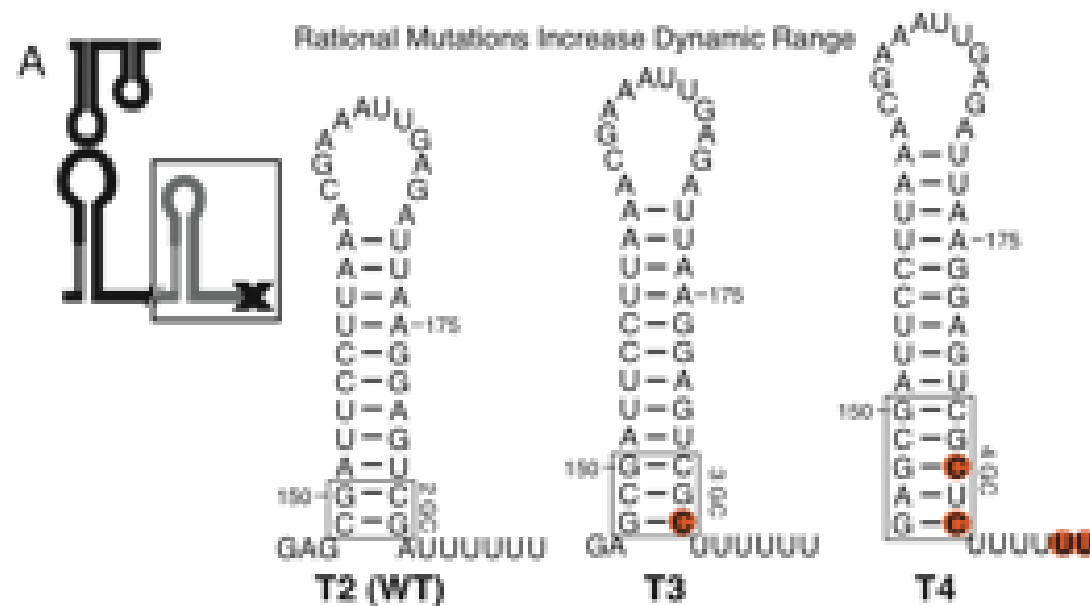


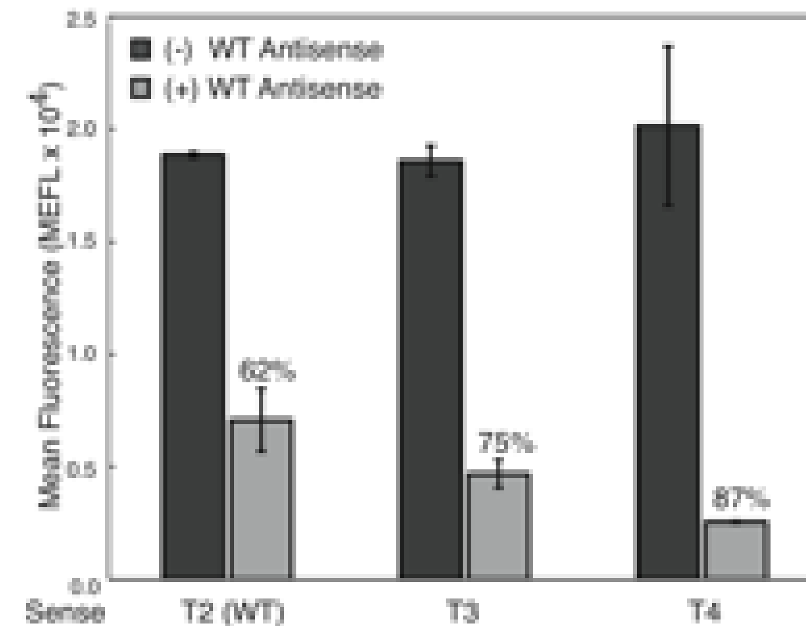
Transcriptional Attenuators: Families

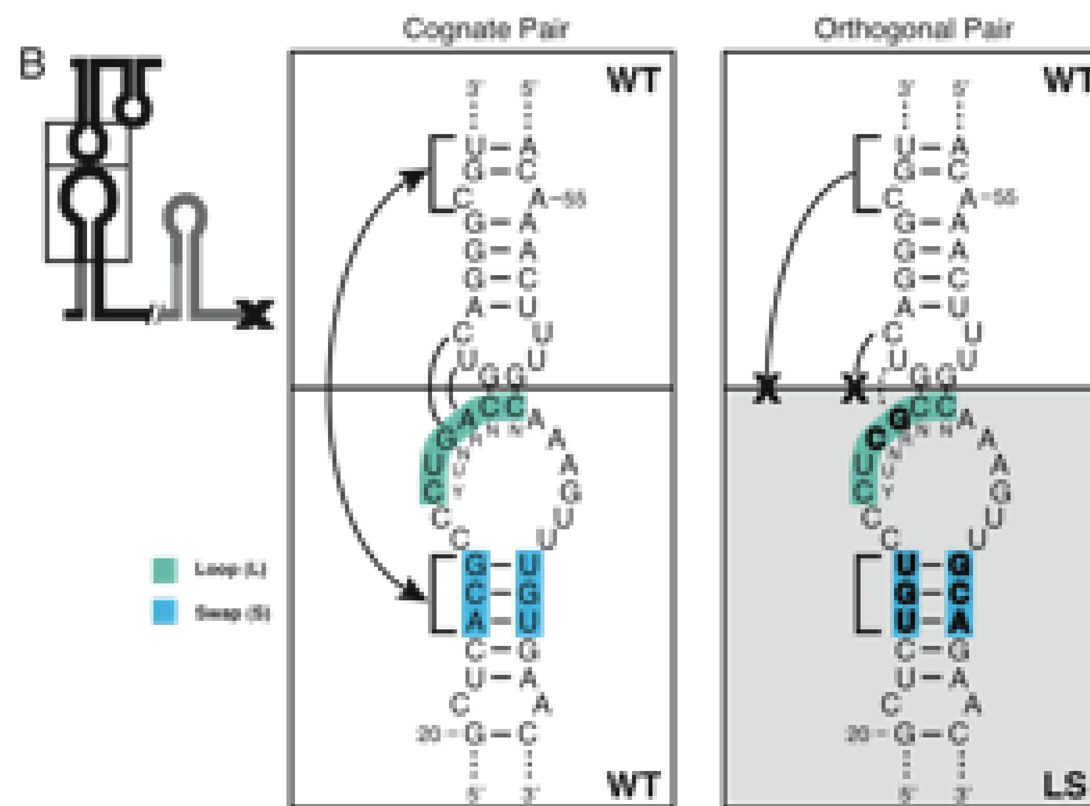
- It is possible to design orthogonal mutant lock-key pairs by introducing mutations into recognition motifs
- Creating family of mutually orthogonal lock-key pairs ready to use
- 4096 possible pairs with this version of the stem-loop.

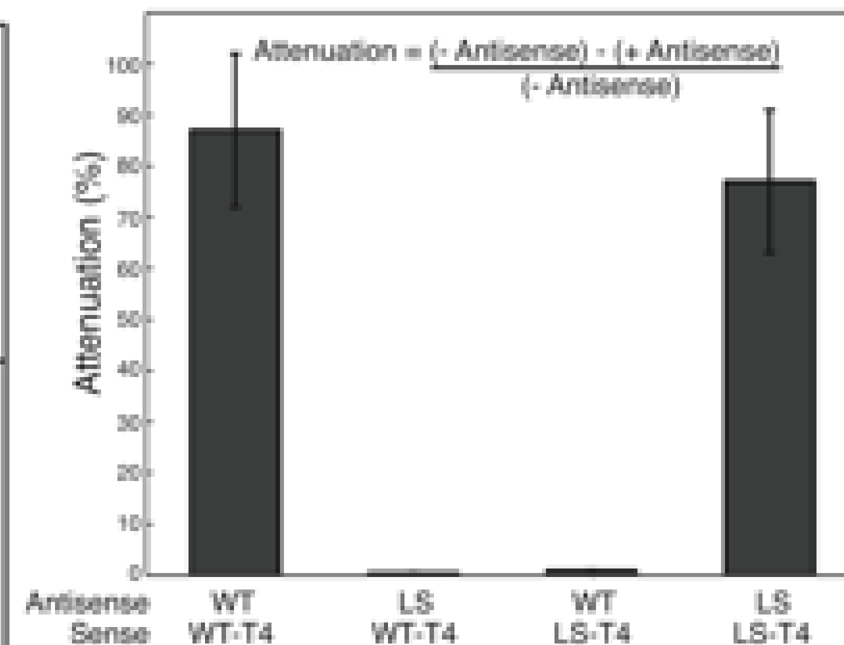


Engineering Dynamic Range and Orthogonality

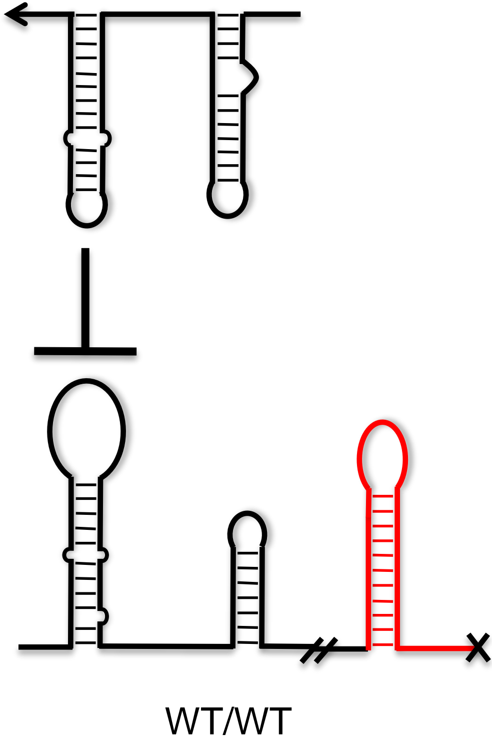


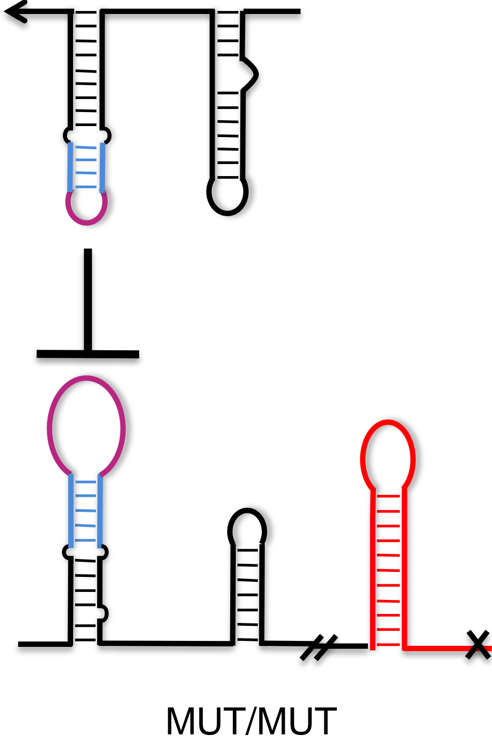




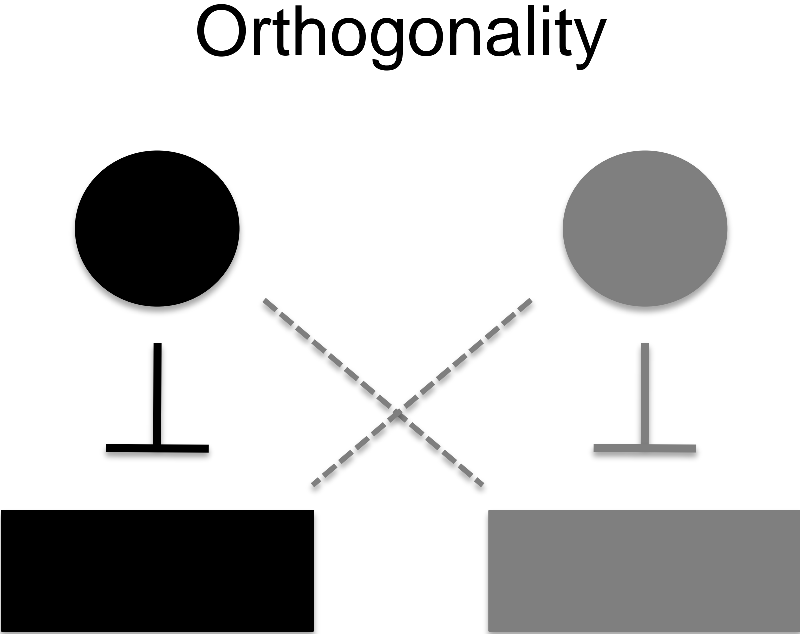


Cracking the Code of Orthogonality





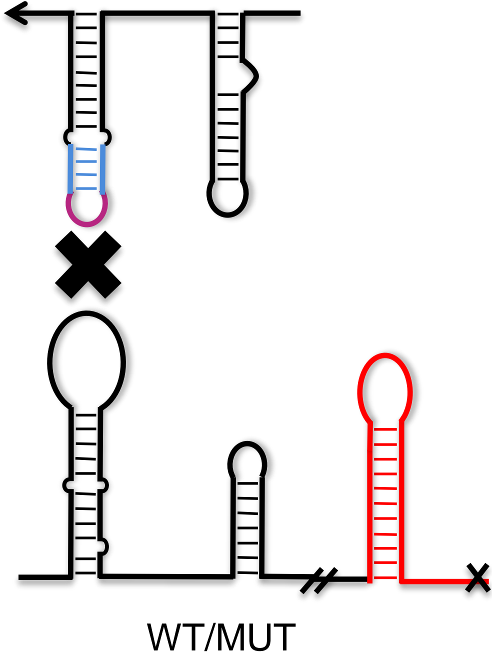
Orthogonality

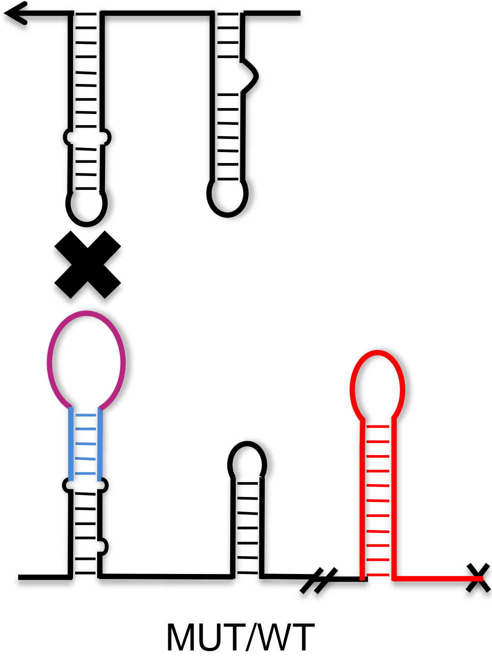


Out of 190nt:

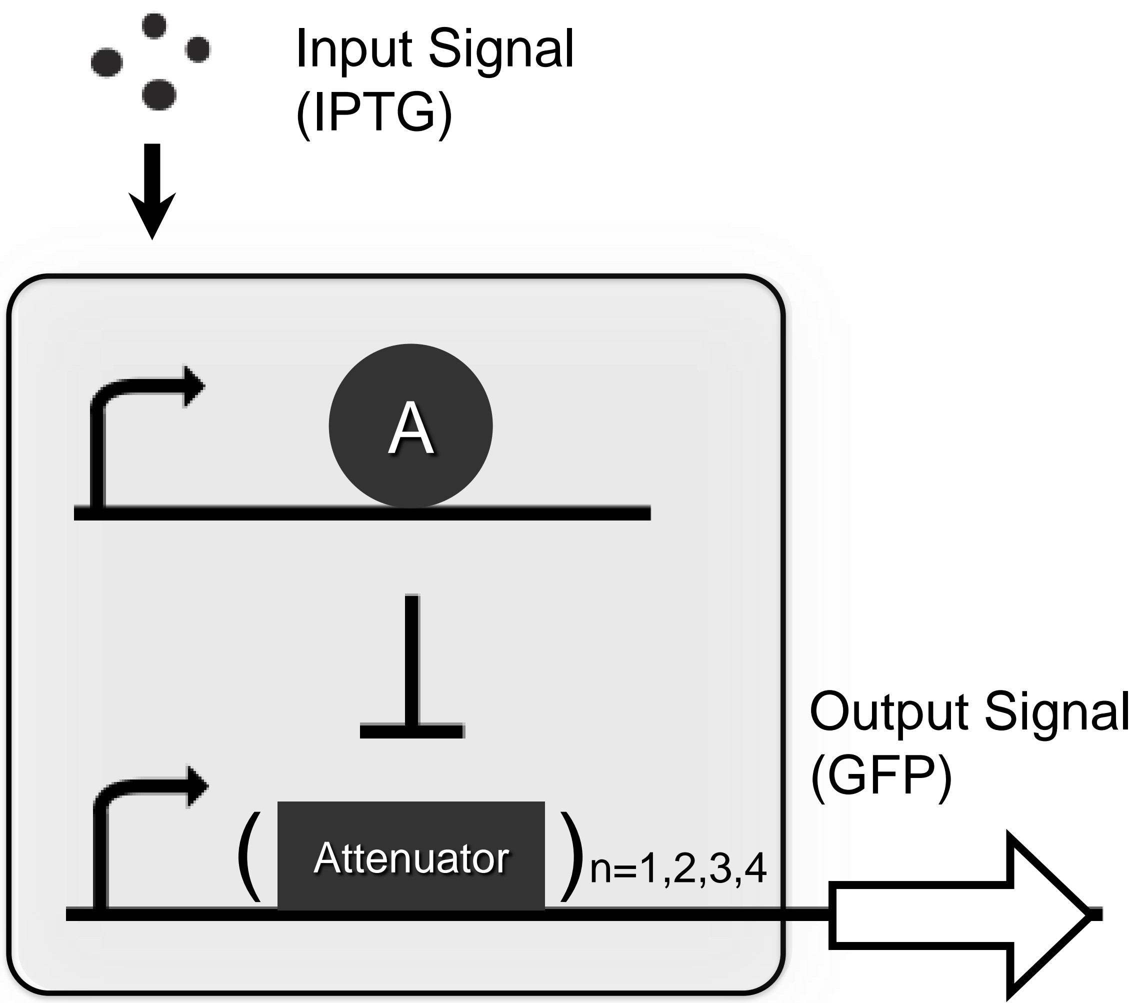
Specificity - 8nt change!

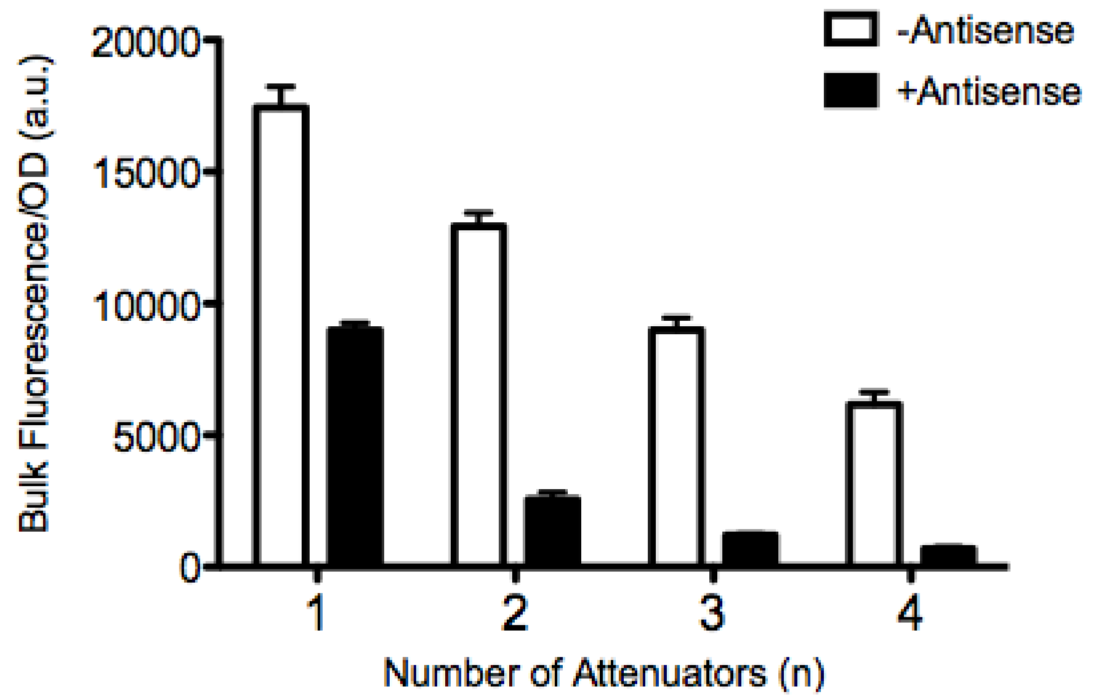
Dynamic Range - 4nt change!



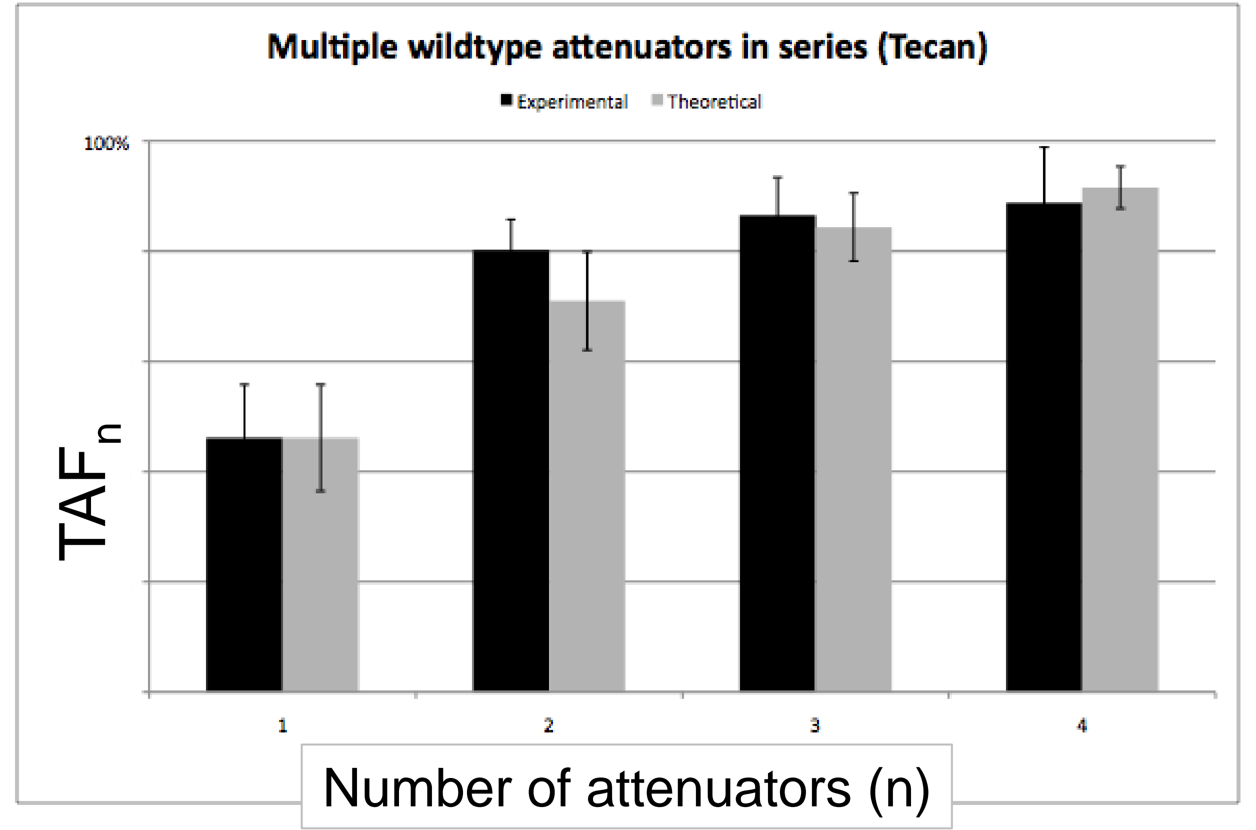


Composing Multiple Attenuators

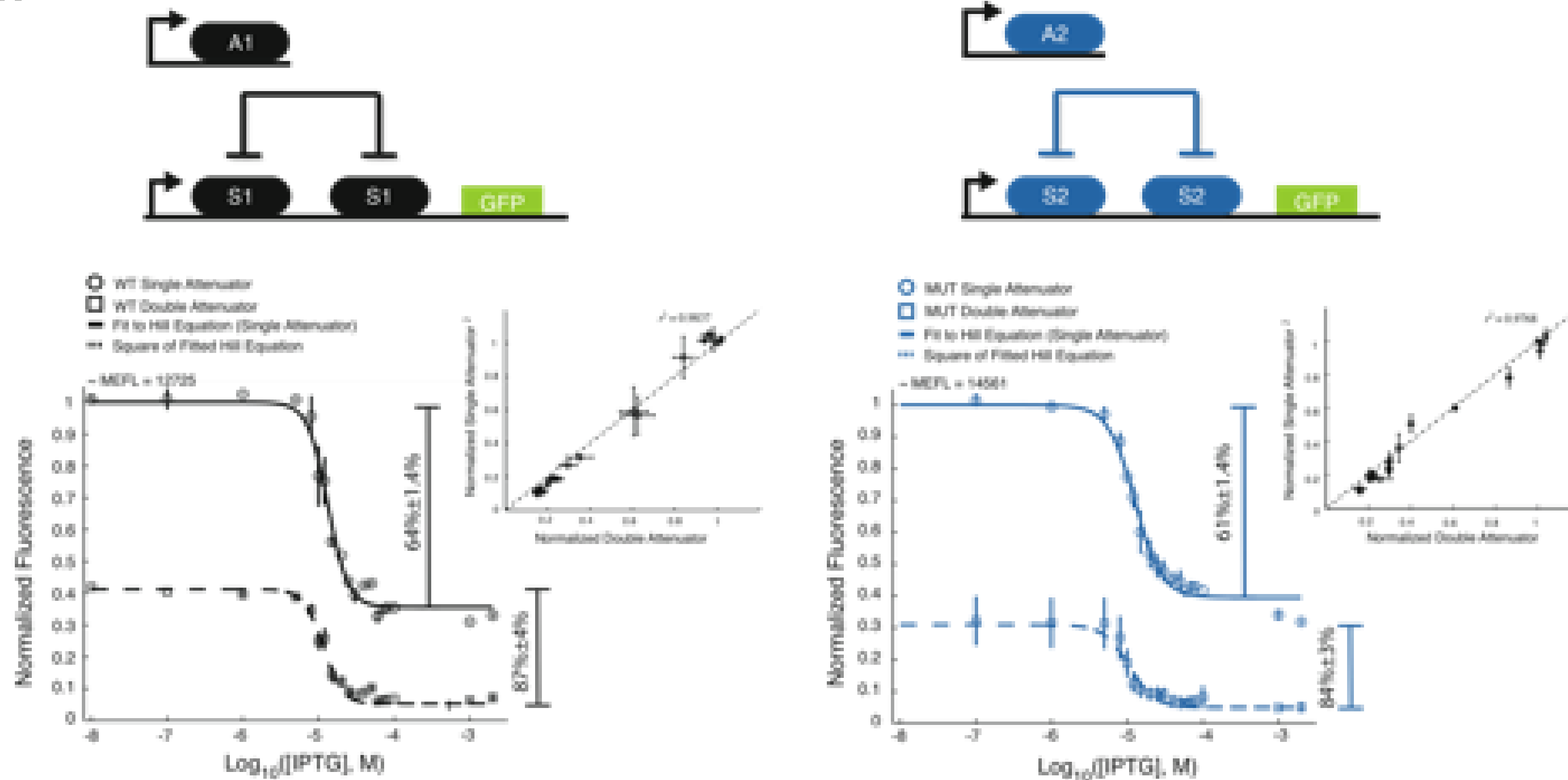


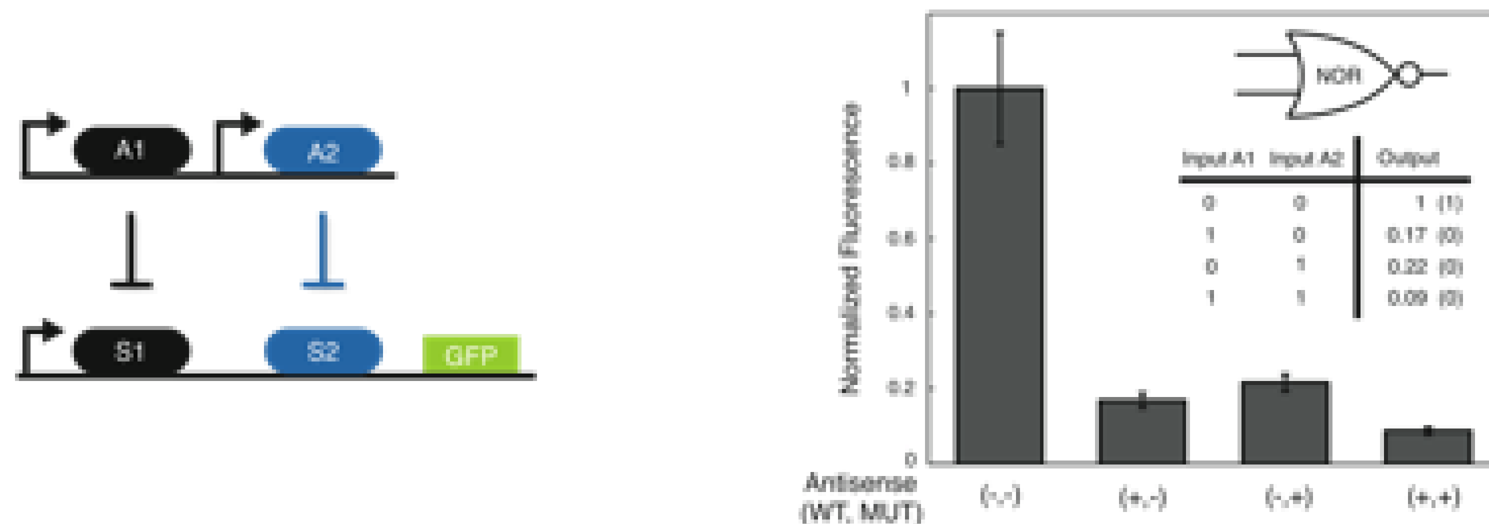


$$TAF_n=1-(TAF)^n$$

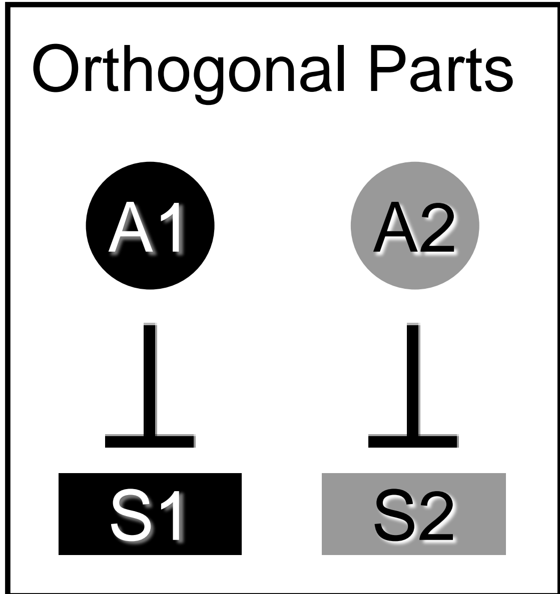


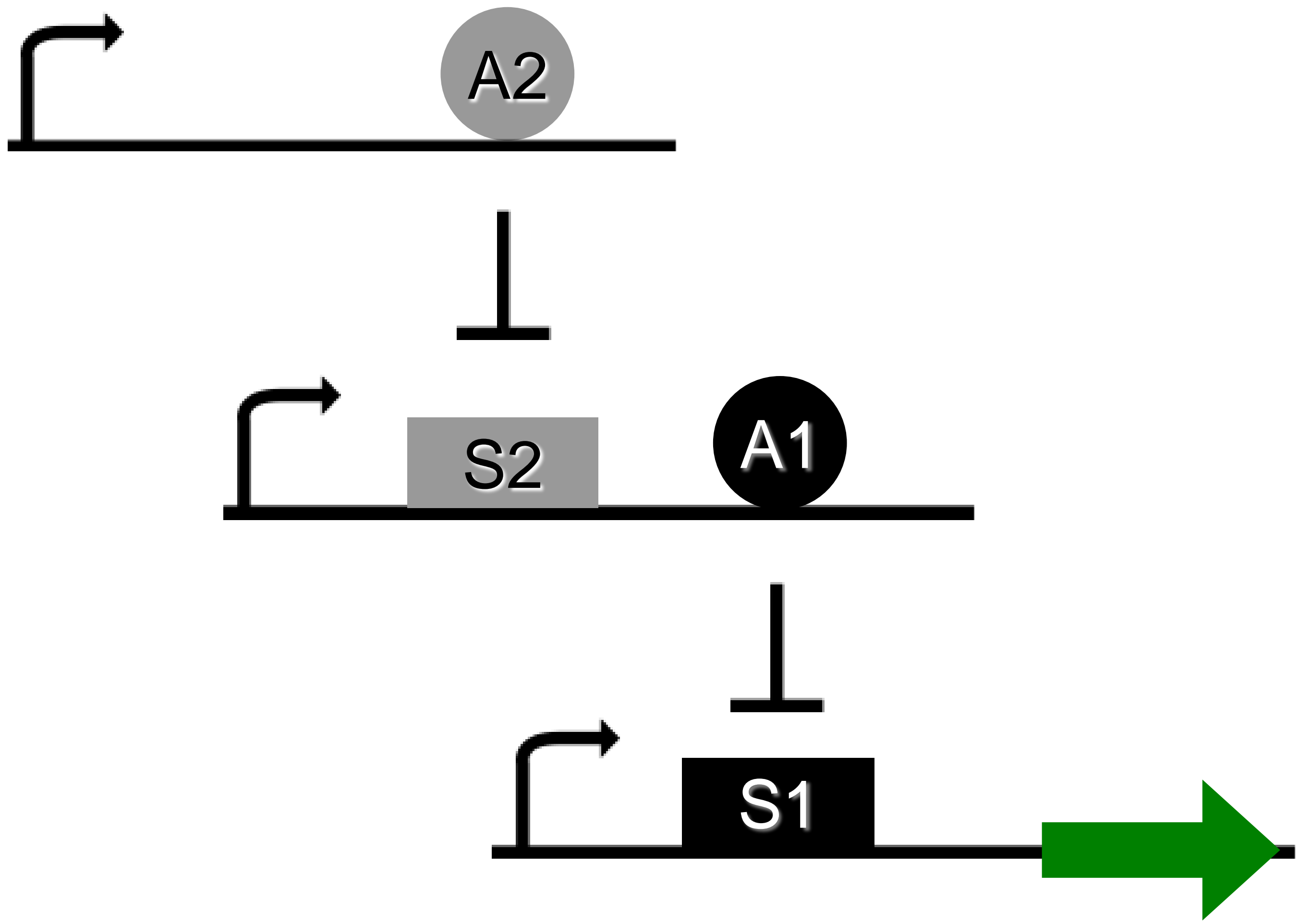
Homogeneous and combinatorial function





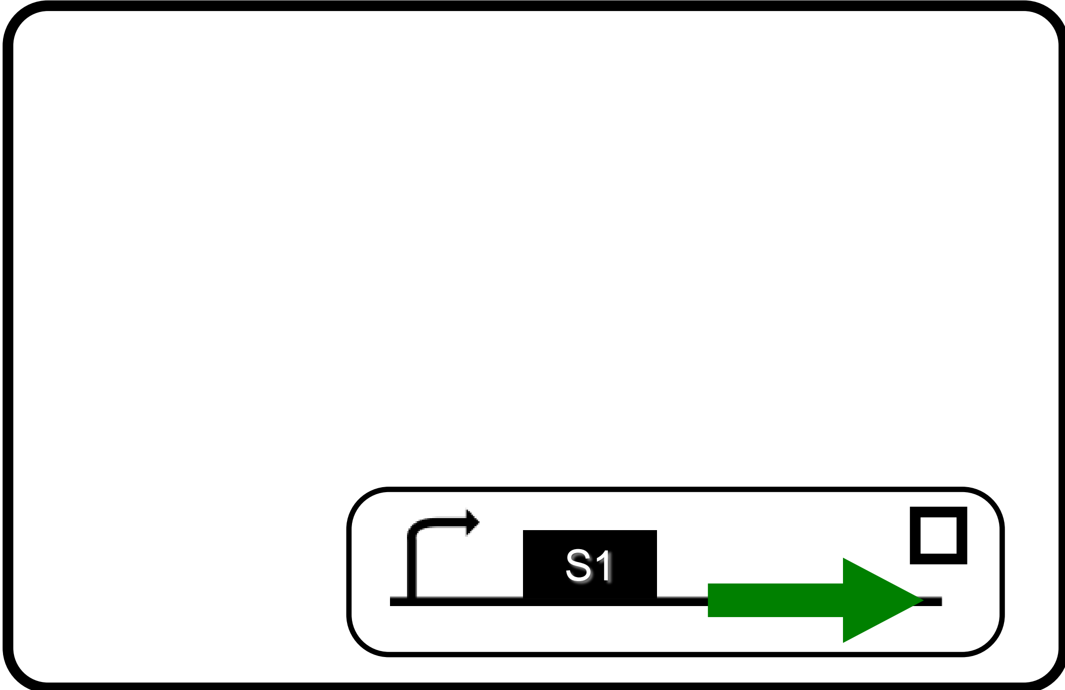
Engineering a 2-Stage Cascade

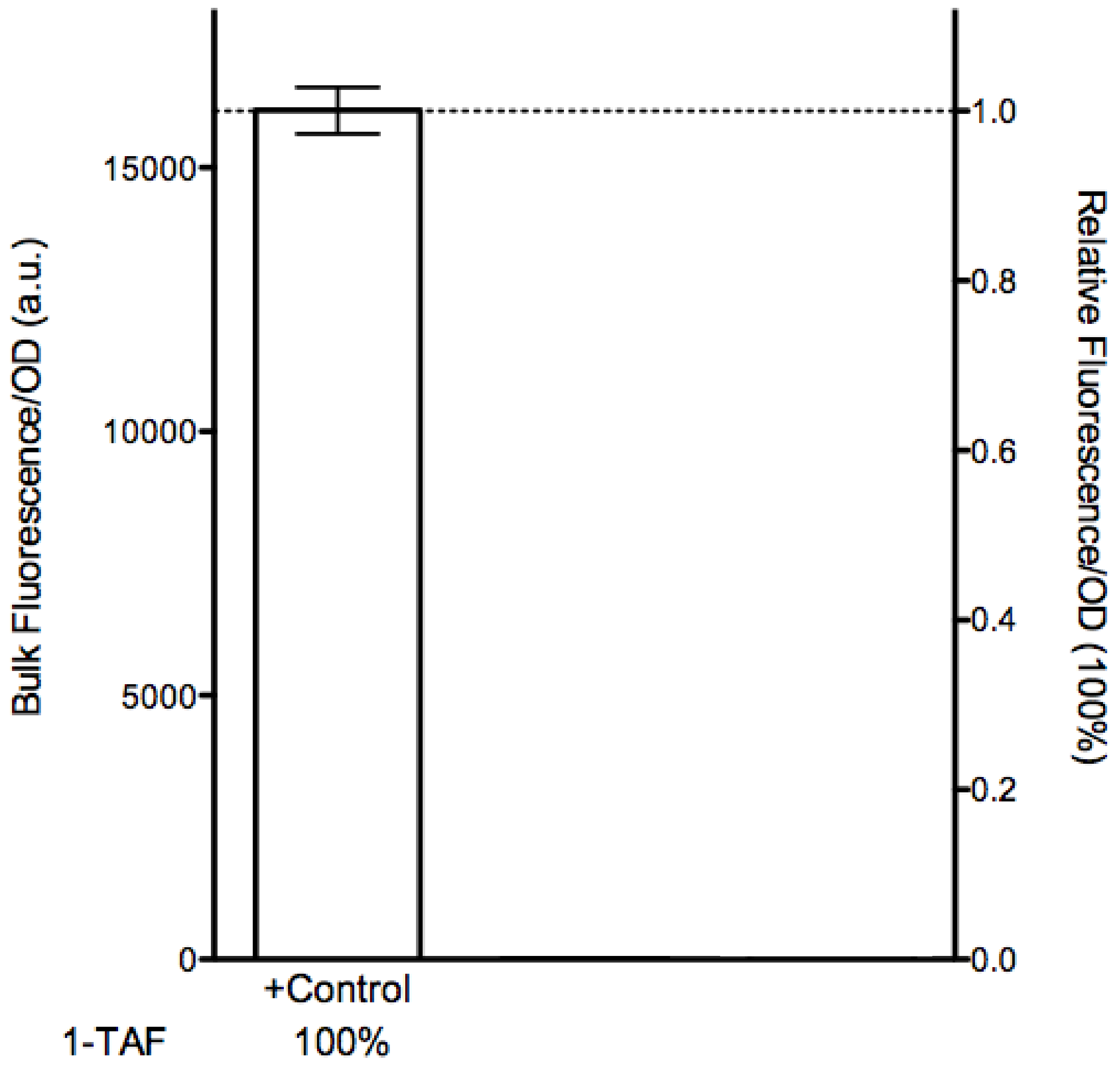




Stanley Qi, Arkin Lab

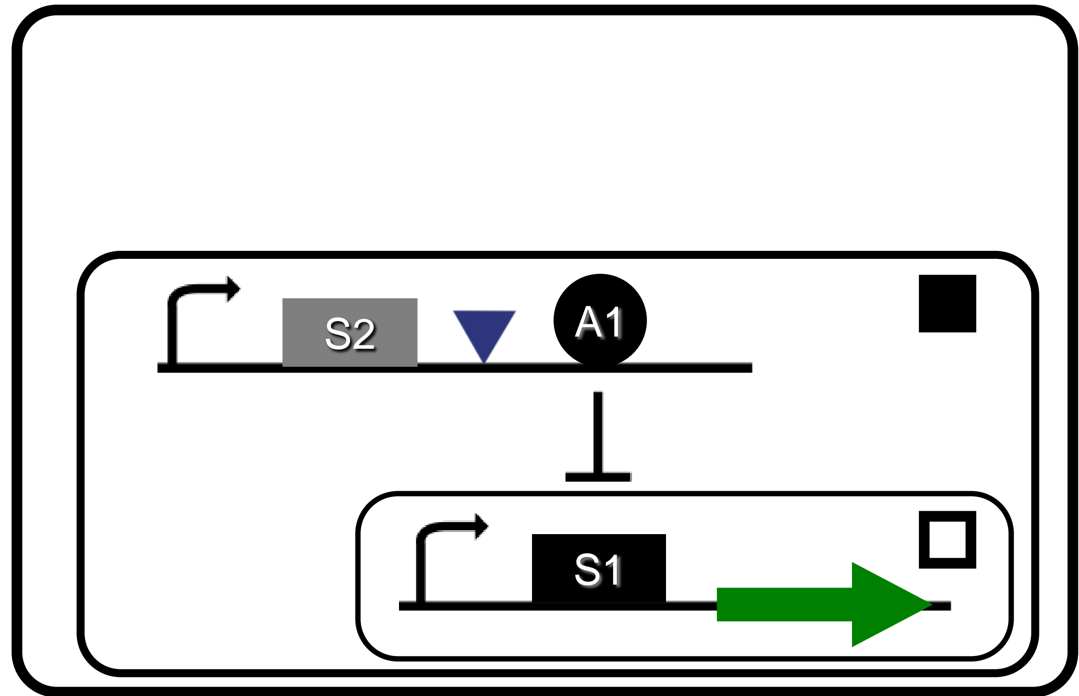
Engineering a 2-Stage Cascade

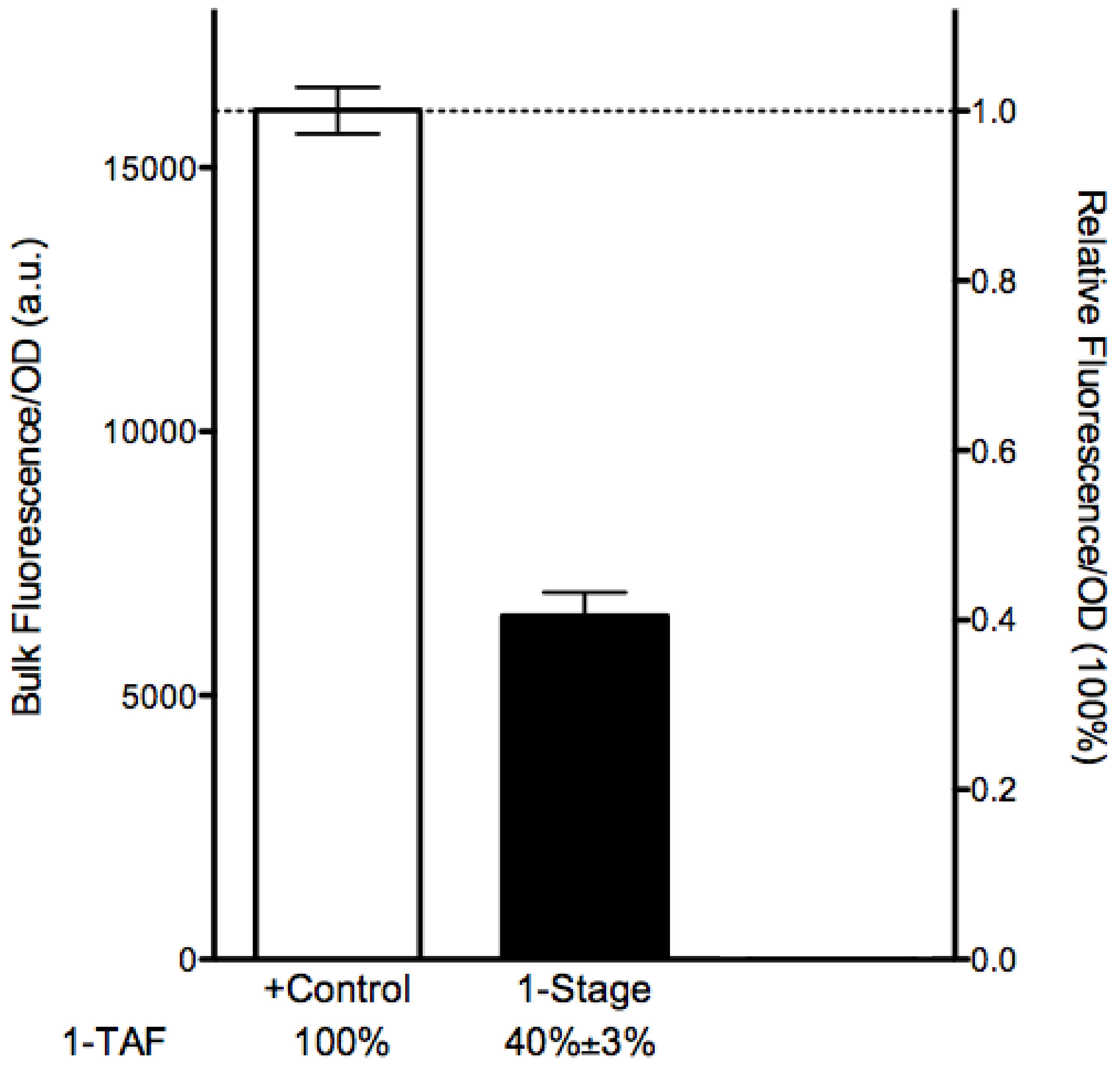




Stanley Qi, Arkin Lab

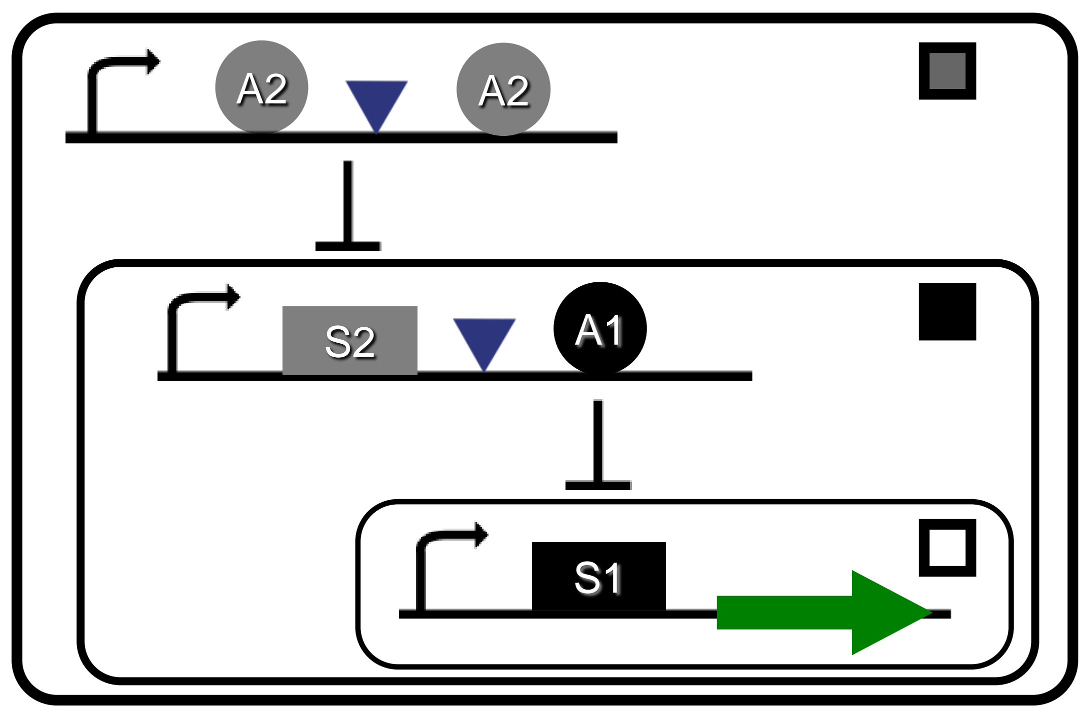
Engineering a 2-Stage Cascade

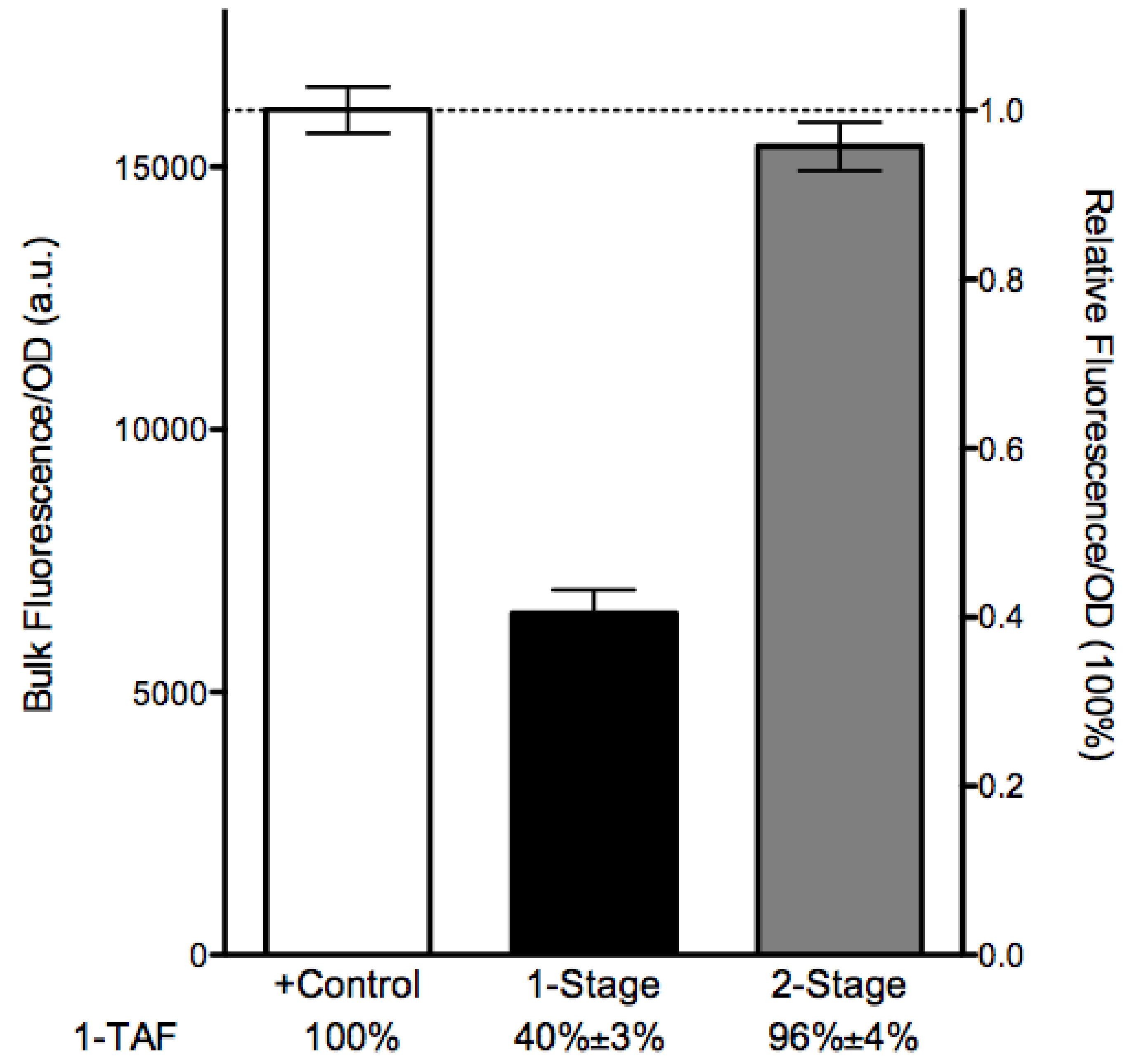




Stanley Qi, Arkin Lab

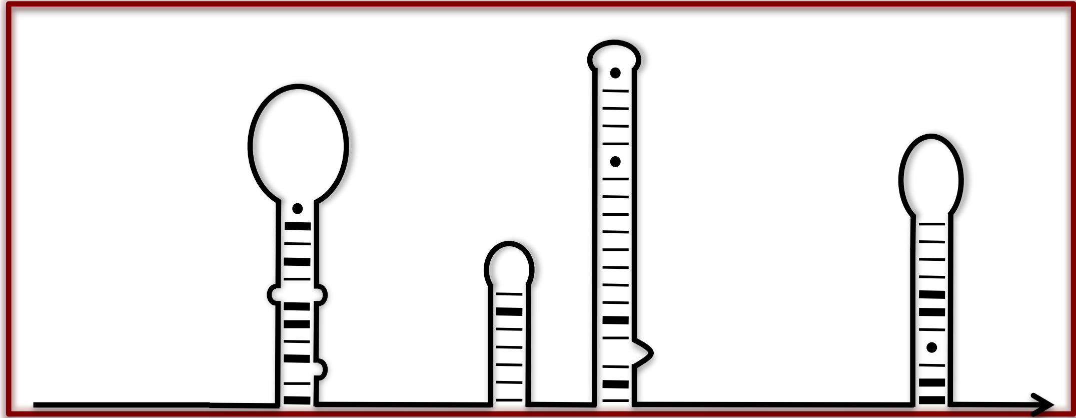
Engineering a 2-Stage Cascade



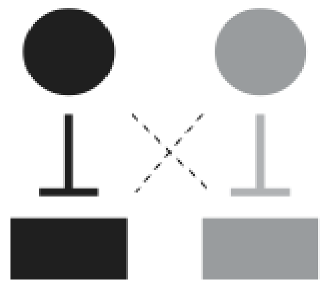


Stanley Qi, Arkin Lab

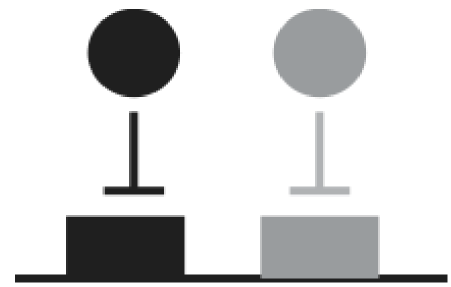
Summary – Engineering Transcription Regulation



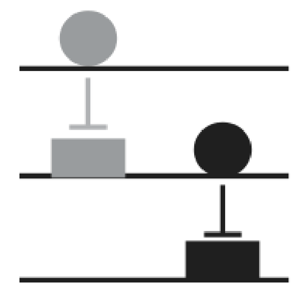
Orthogonality

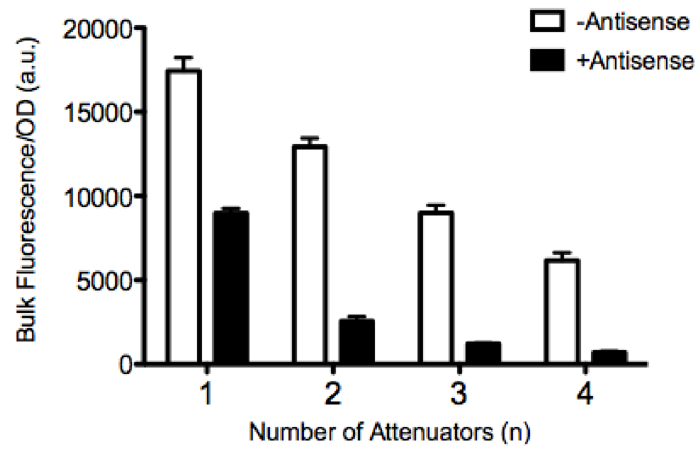


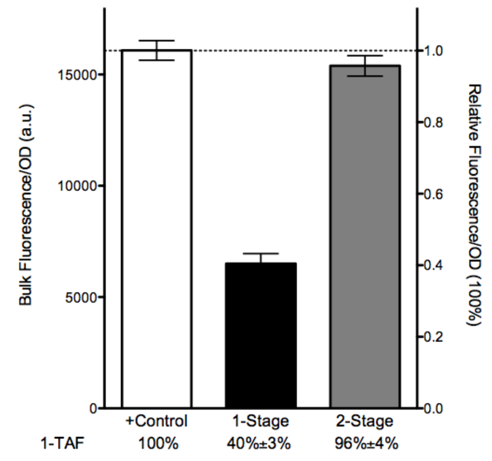
Composability



Connectivity

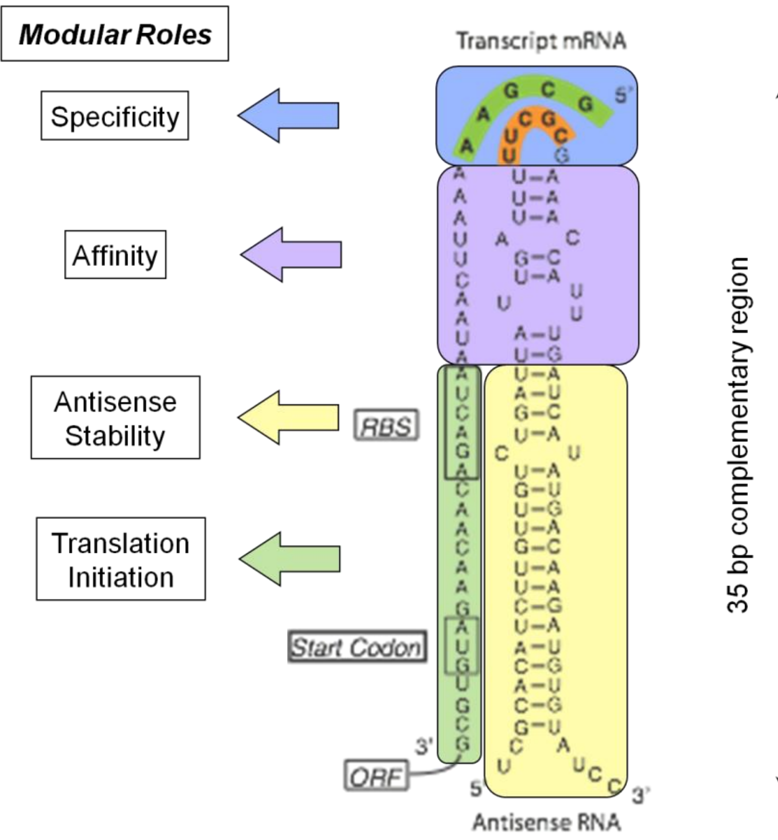


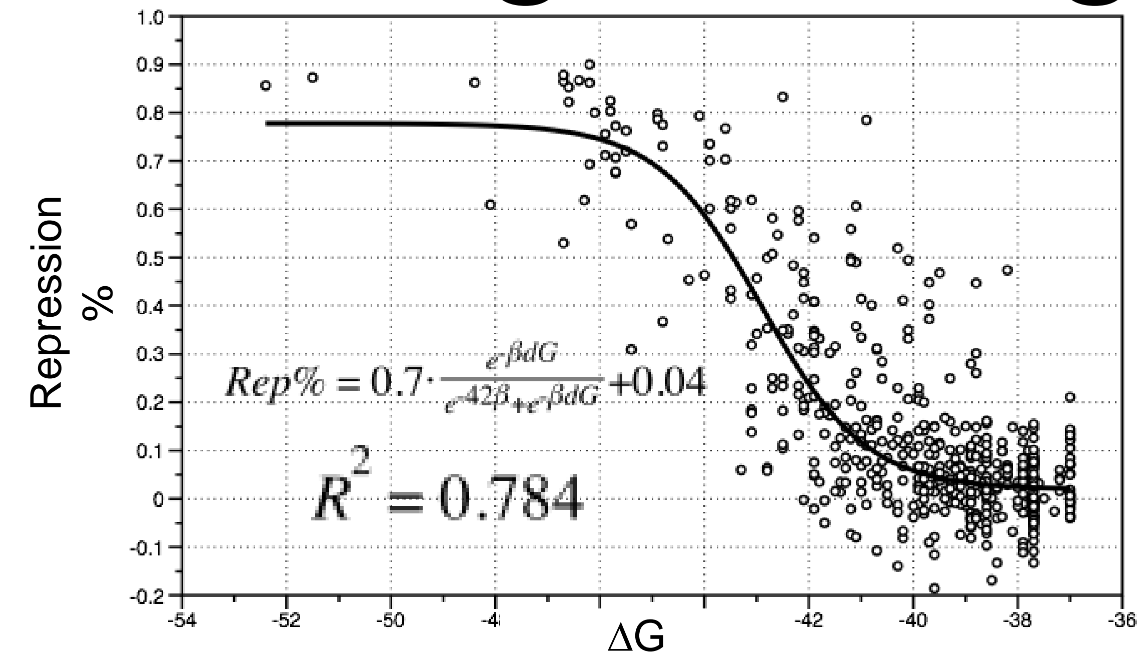




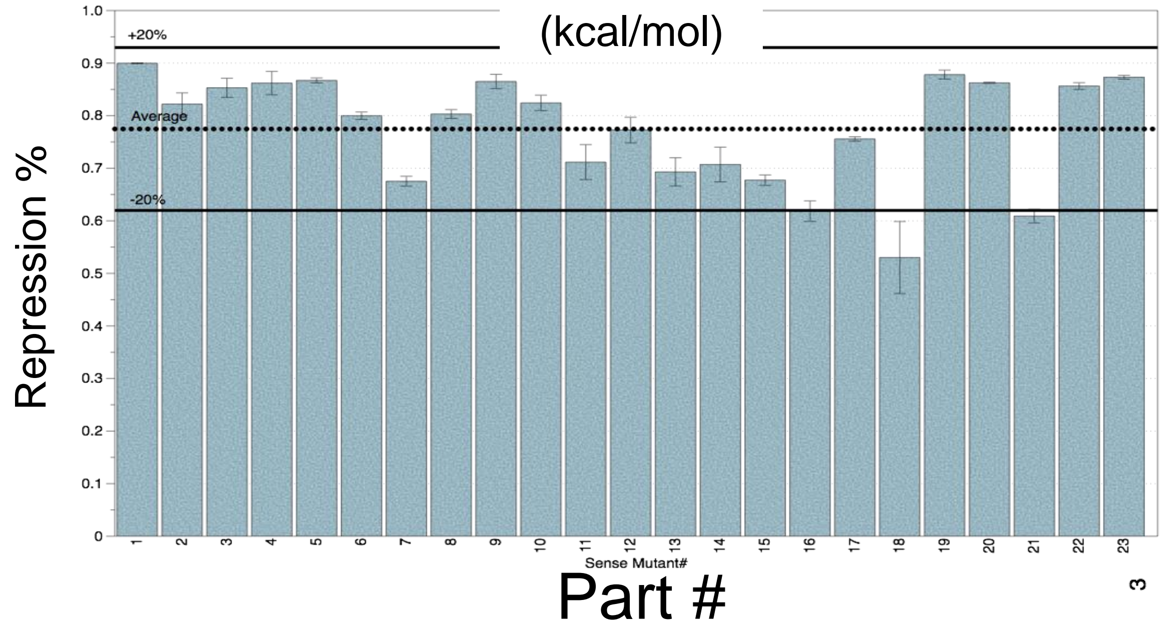
arkin laboratory
of systems and synthetic biology

Translation engineering





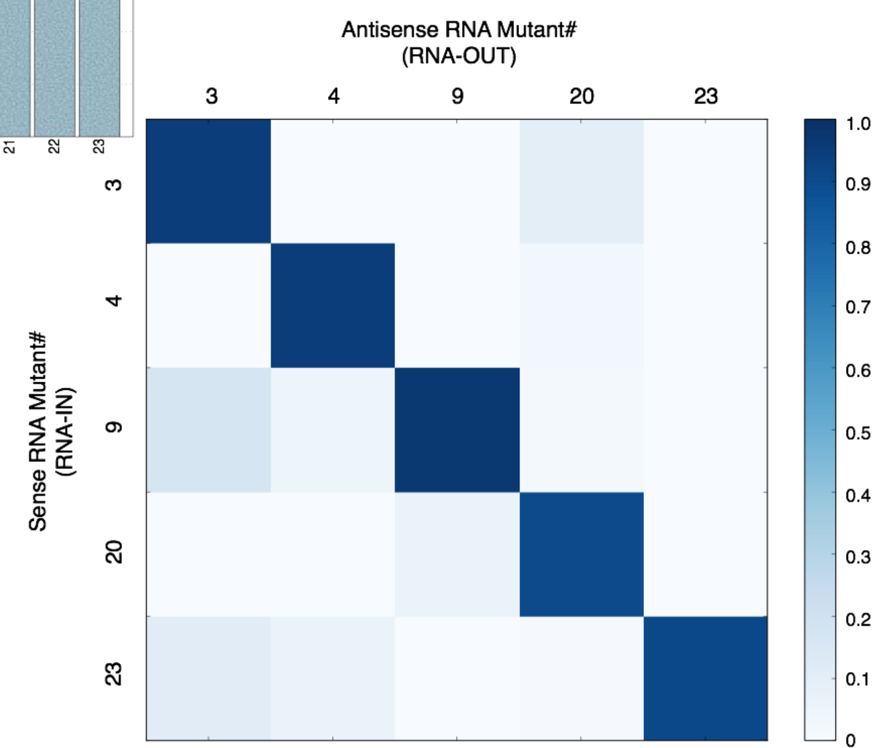
Predictive design



Designed homogeneous operation

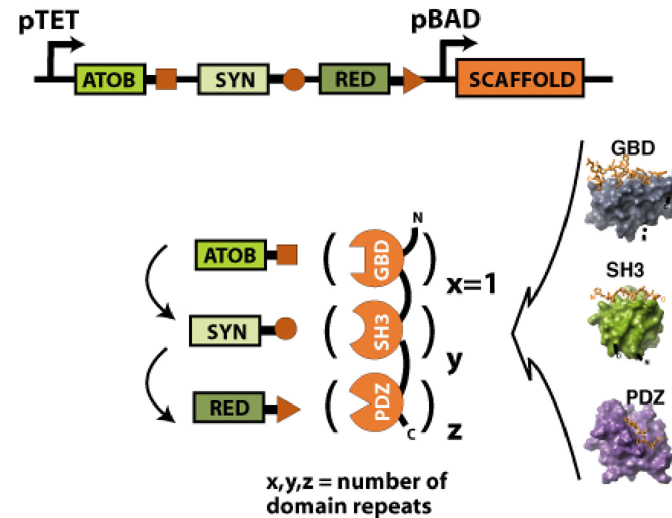
- RNA IN/OUT
- >4096 possible pairs with expanded specificity region
- Modular/composable molecular function

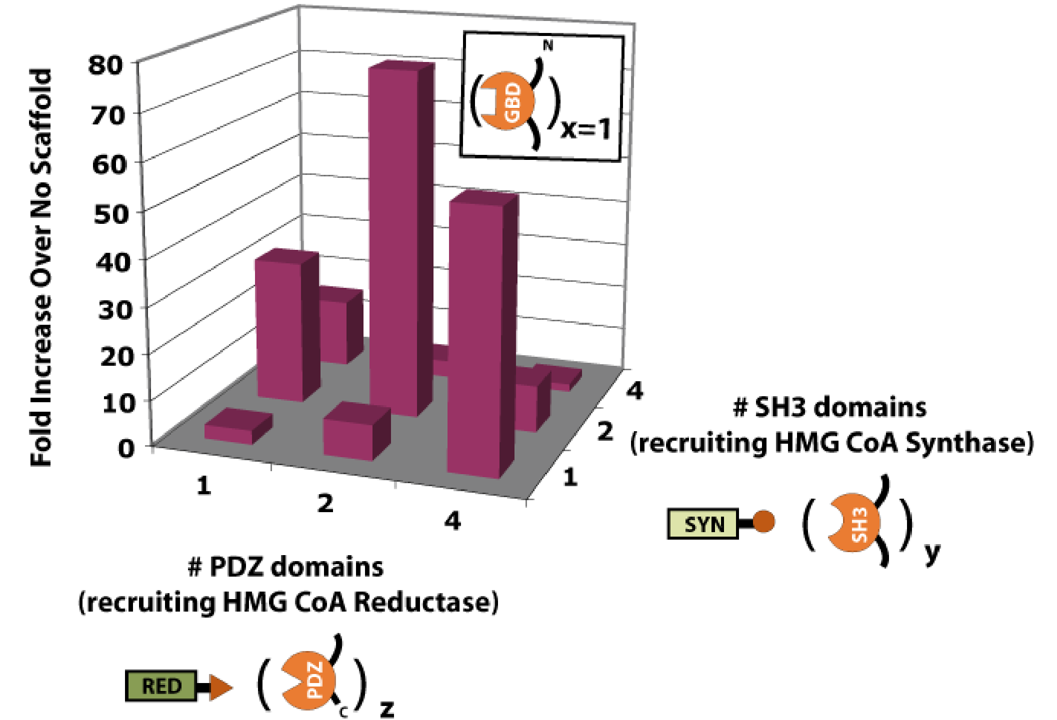
Designed orthogonality



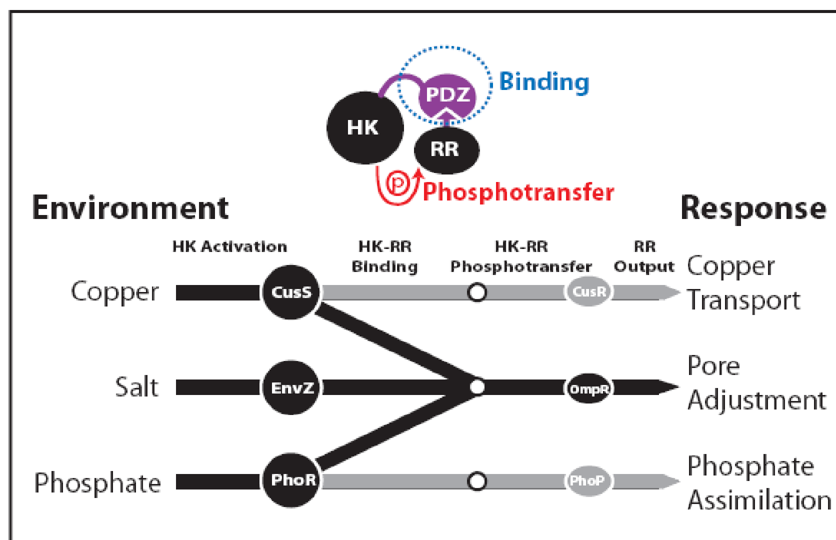
Modular Control of Signal and Carbon Flux

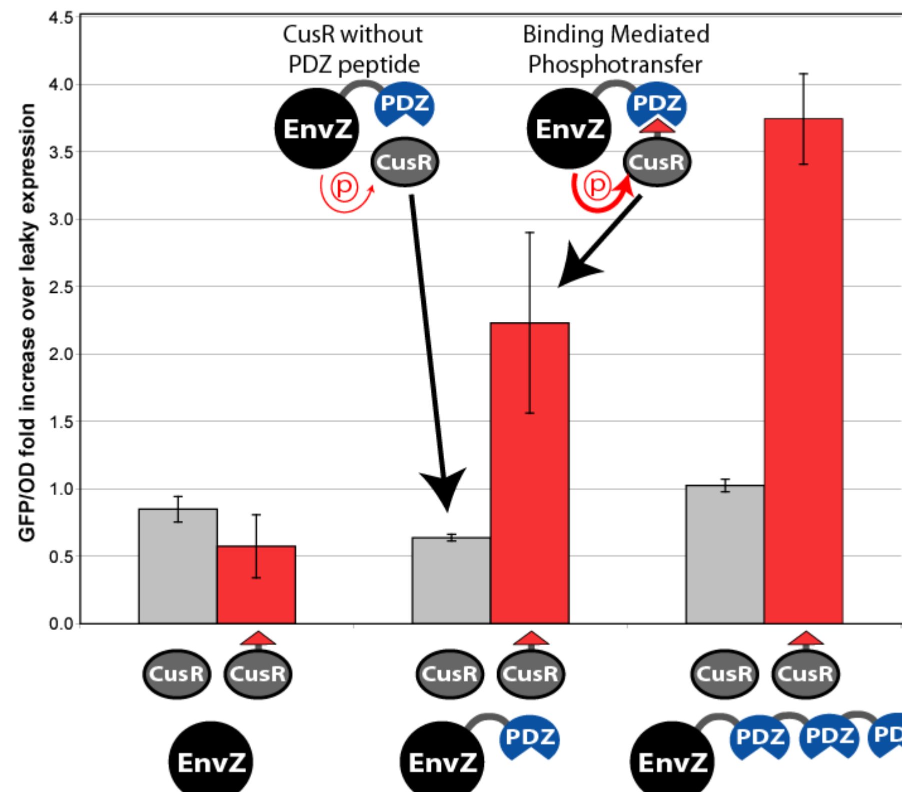
Optimization of metabolism (Dueber/Keasling)

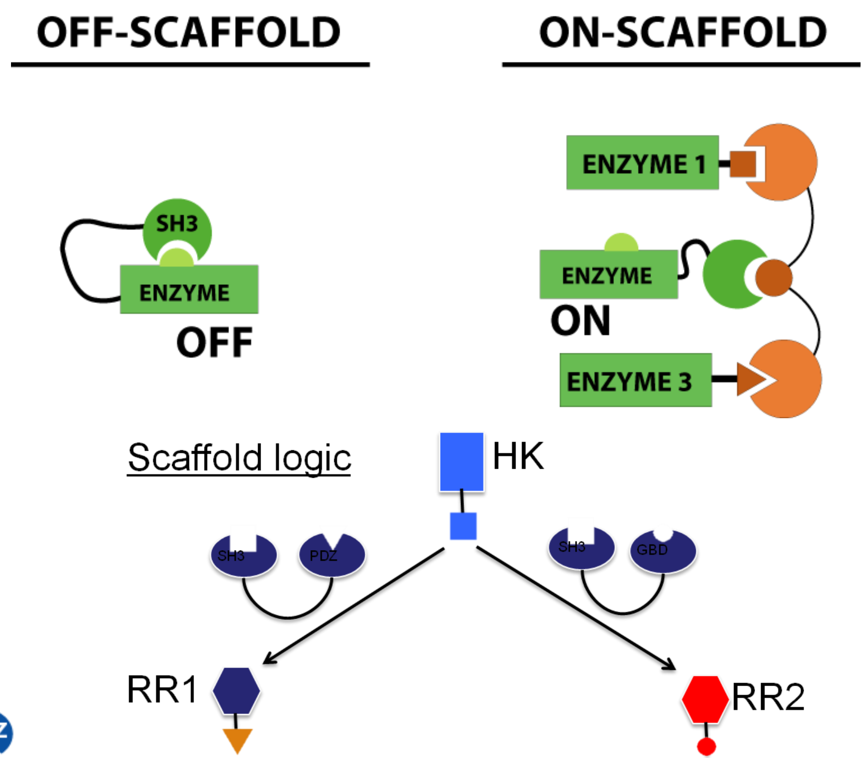




Control of Signal Transduction (Whitaker, Dueber, Arkin)



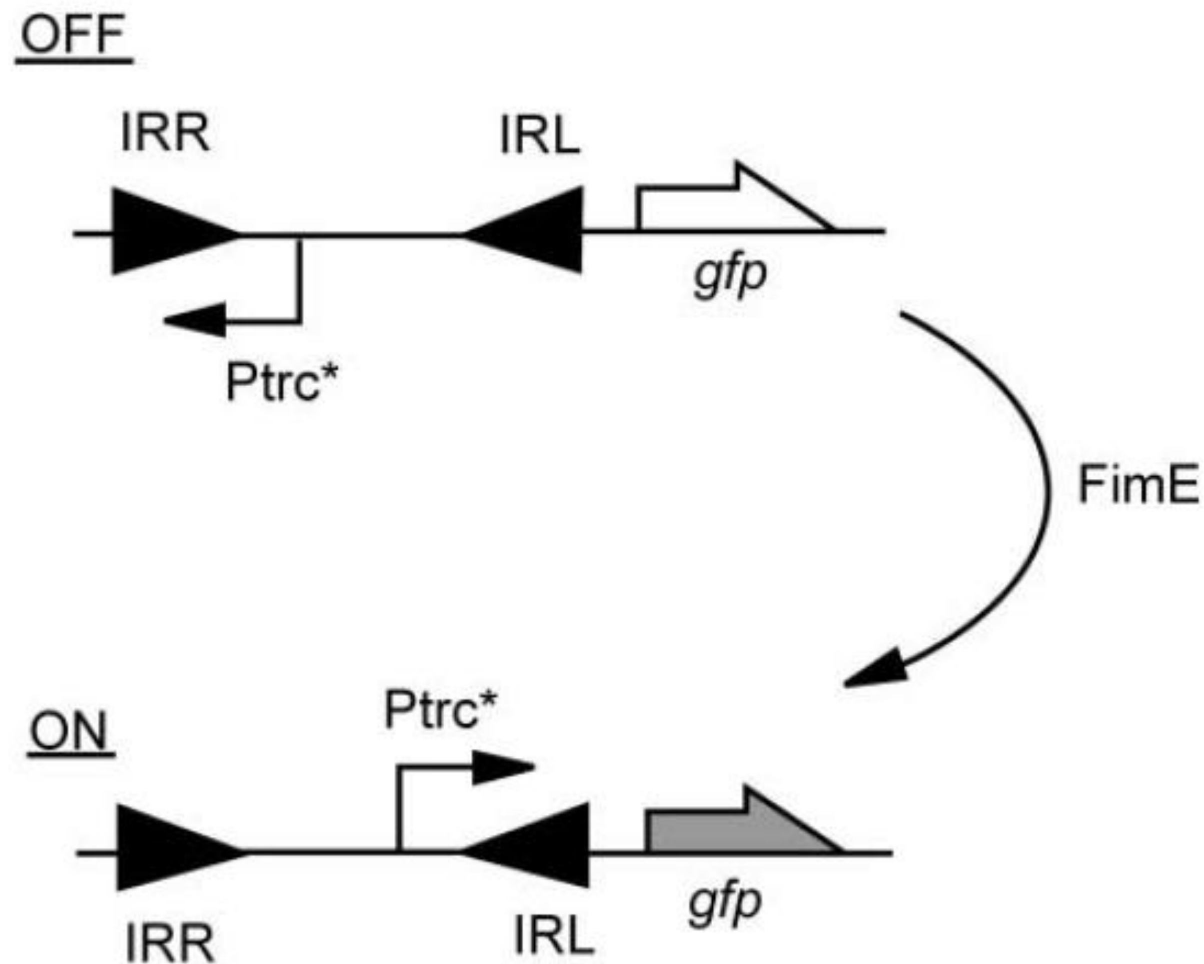




DNA Write

Tim Ham, **Michael Samoilov**

Recombinases



Invertases: the Fim system

- Interesting operations
- Inversion
- Insertion
- Excision

- Nearly Boolean

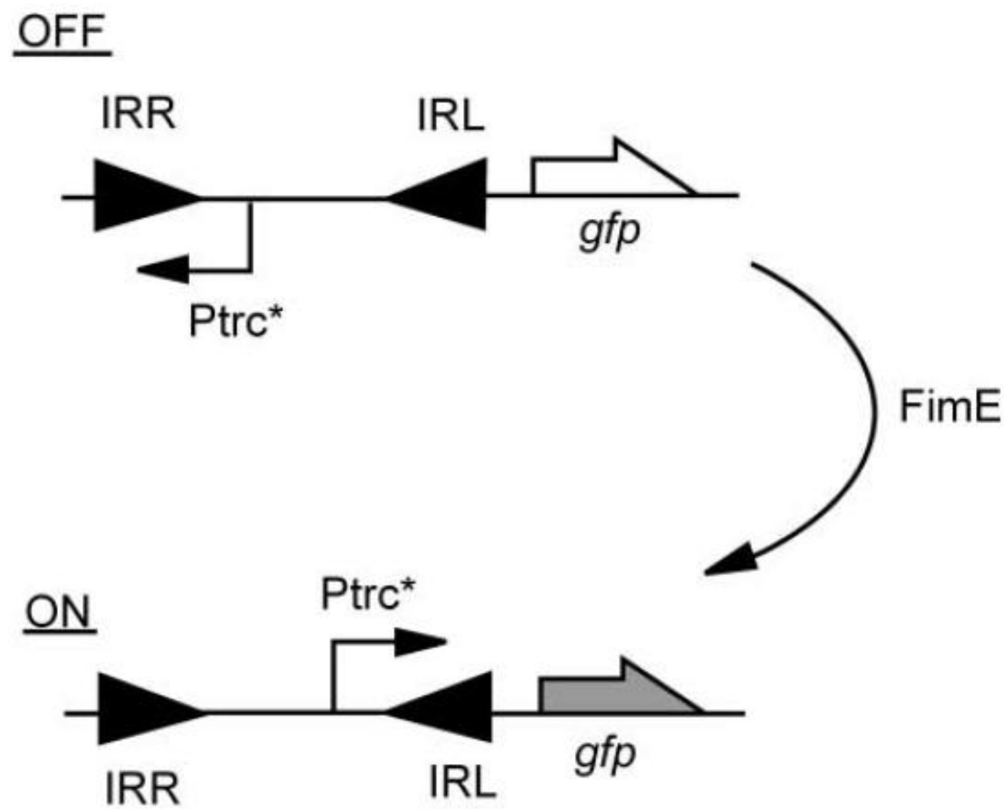
- State accessible after death and transmissible between cells

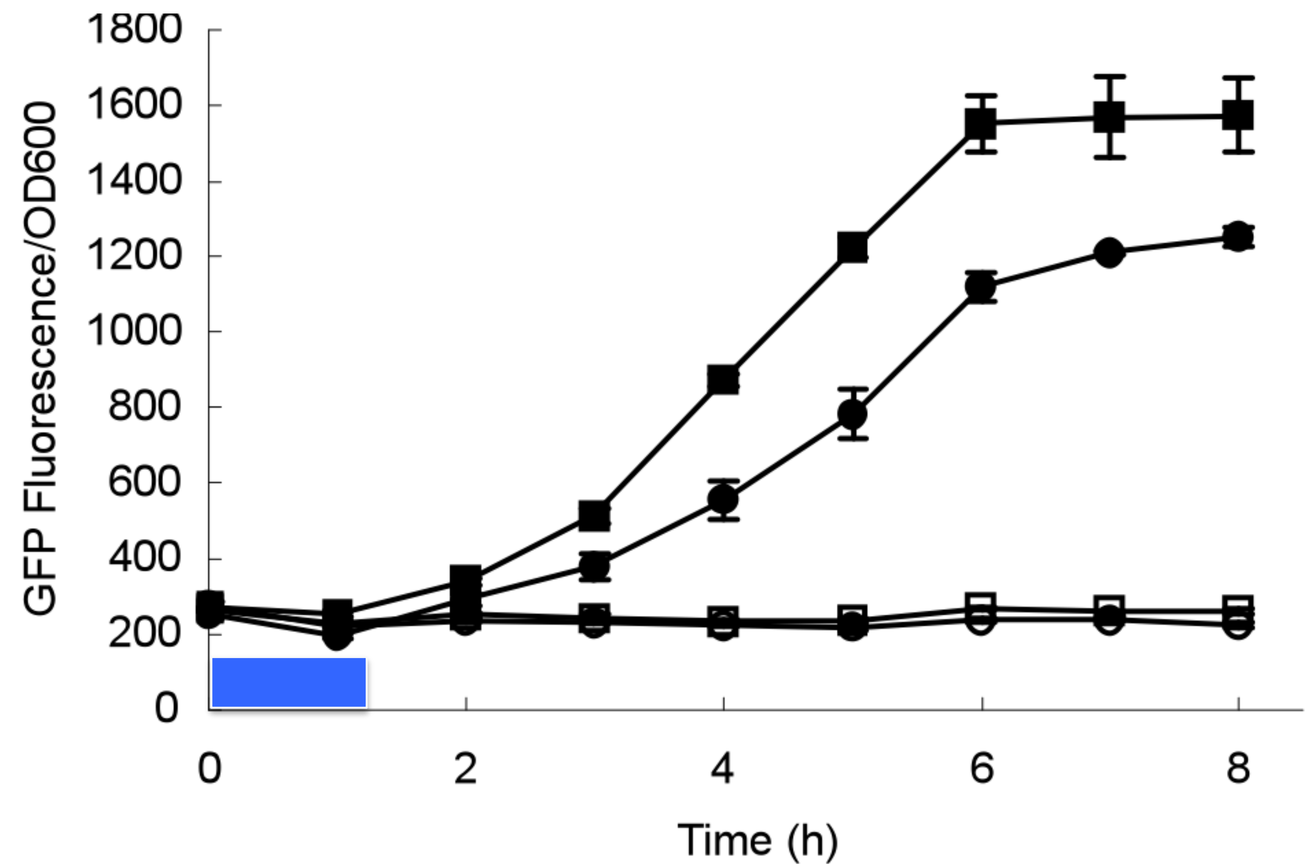
- Geometrically programmable?

- Possible huge state space for logical machines encodable in relatively little DNA

- Flexible acceptance of active elements in internal regions.

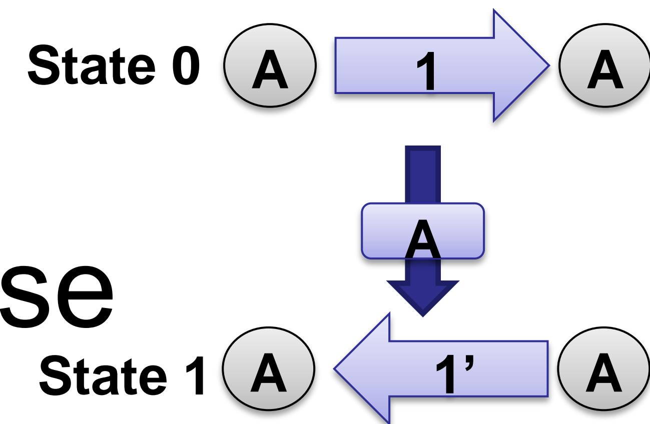
Invertase dynamics



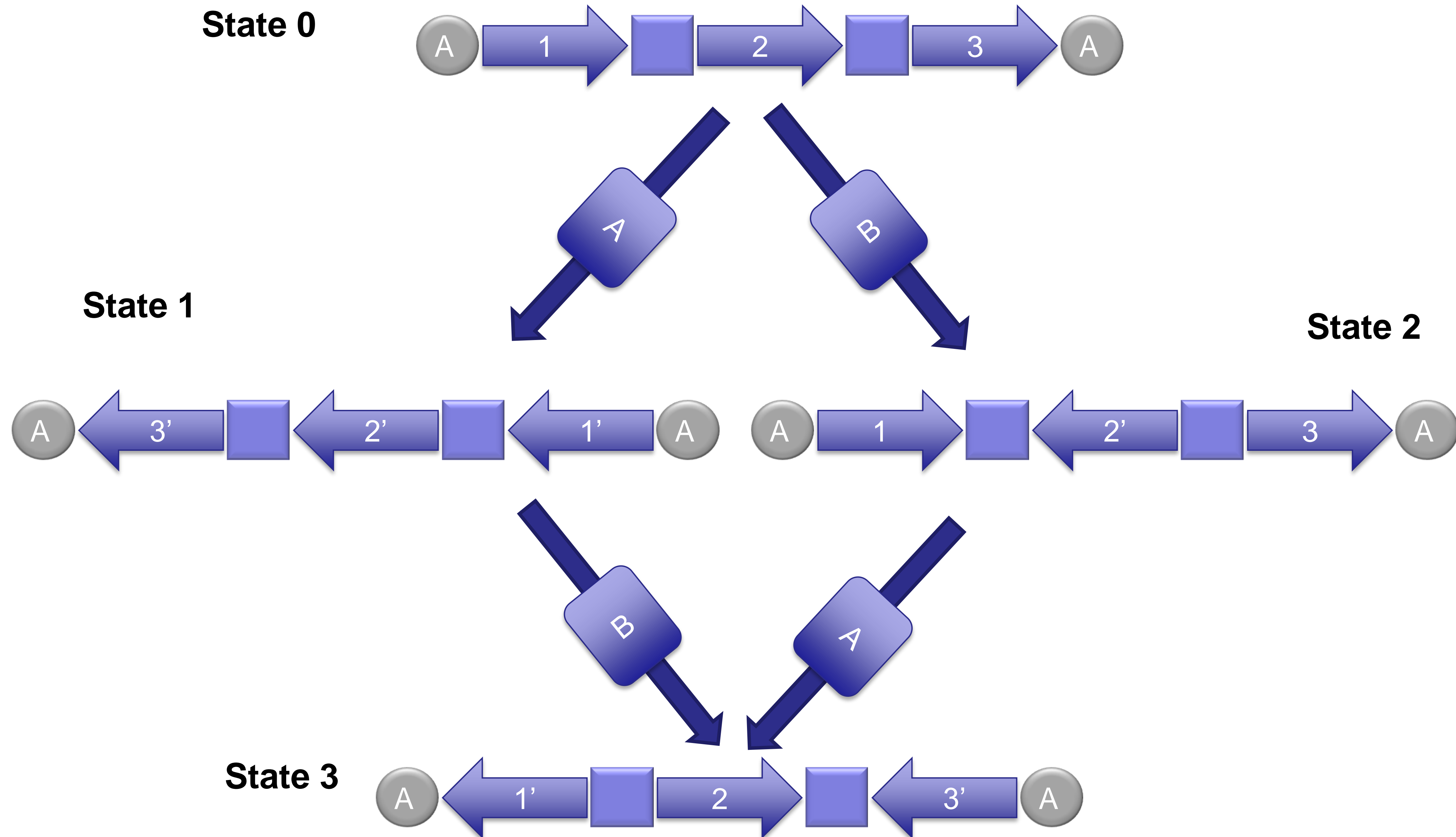


Leakless

System can hold "state" after pulse

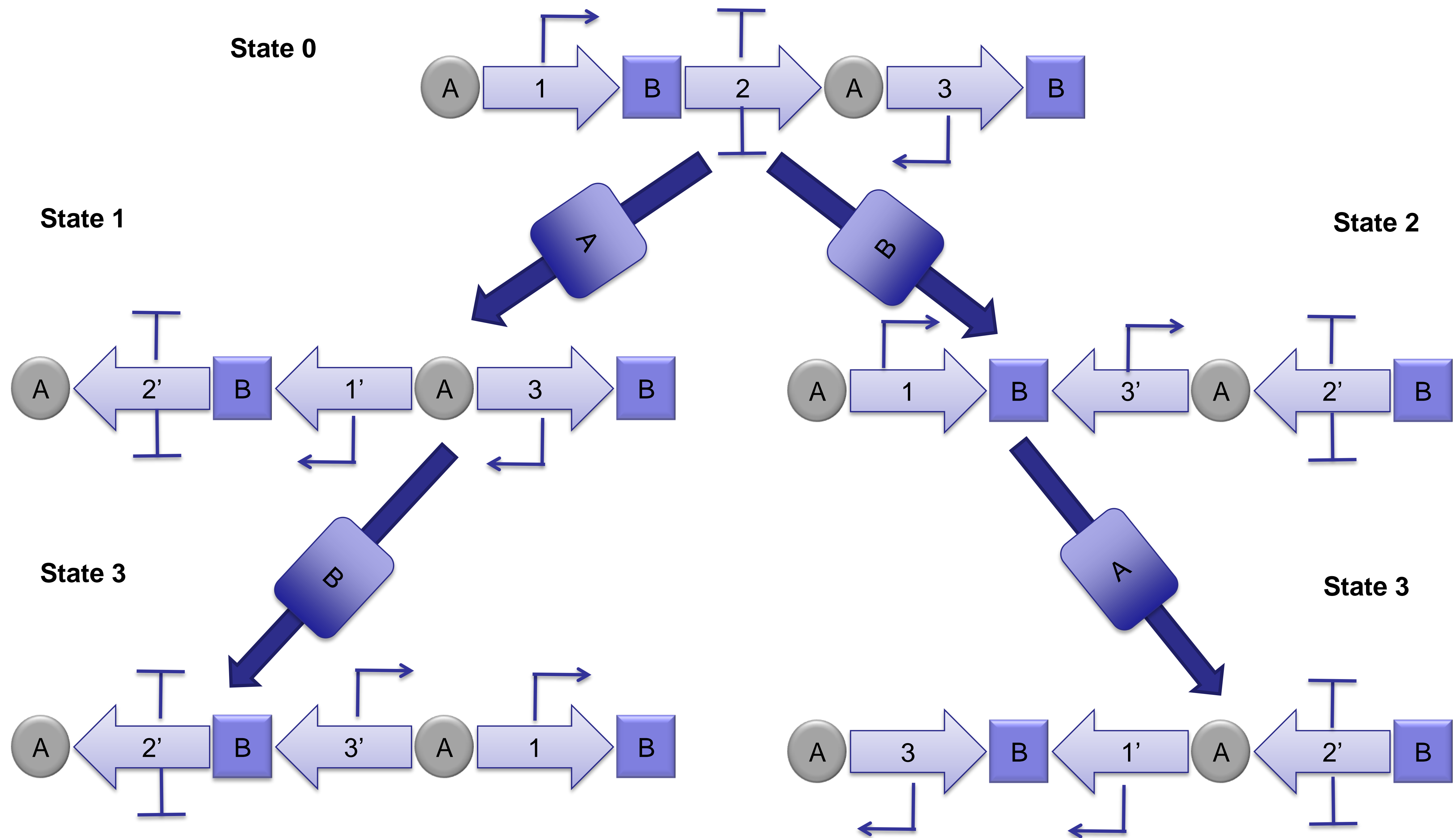


Placement of two invertase sites.



N! input sequences

Placement of two invertase sites.

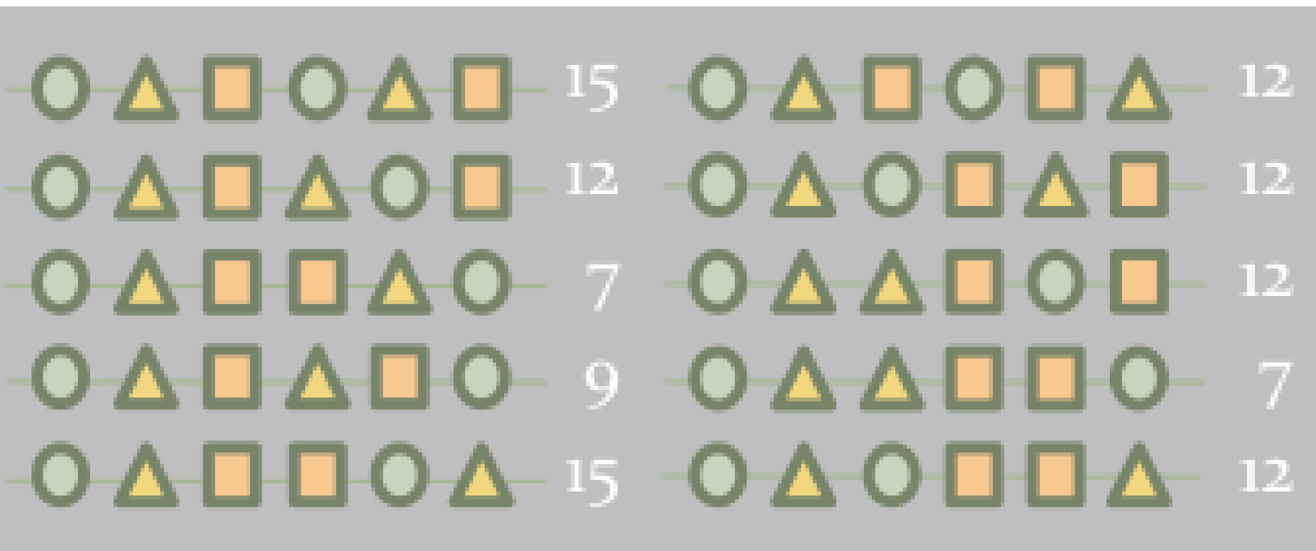


$$S(N) = \sum_{k=1}^{N} k! C(N,k)$$

Maximum number of states available with N pairs of sites

Number of devices and state space grows rapidly with N.

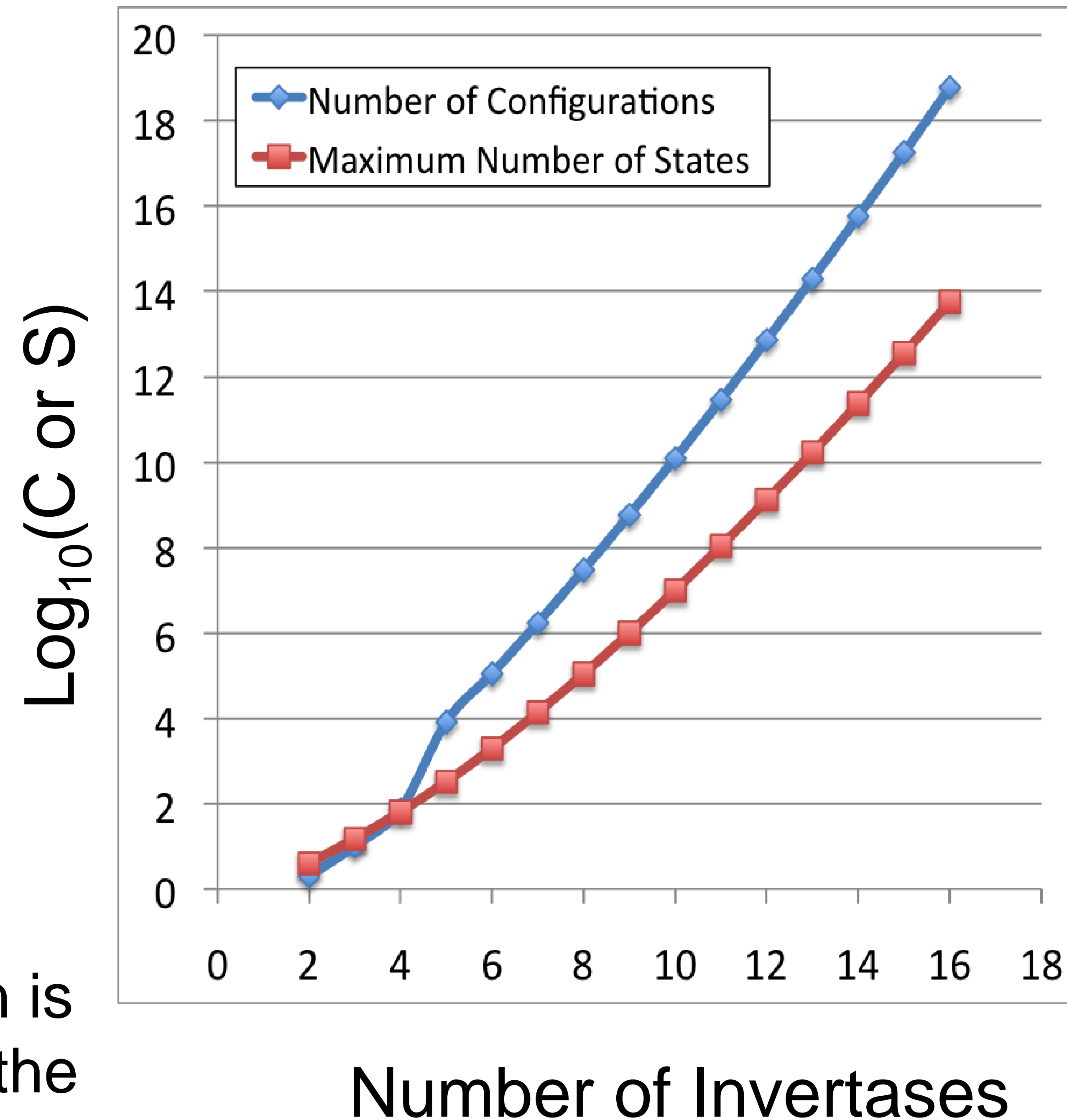
Configurations of 3 site pairs



For n=N-1 there are:

$$a(0)=1;\ \text{for } n>0,\ a(n)=(2n-1)!!-\sum_{k=1}^{n-1}(2k-1)!!\,a(n-k)$$

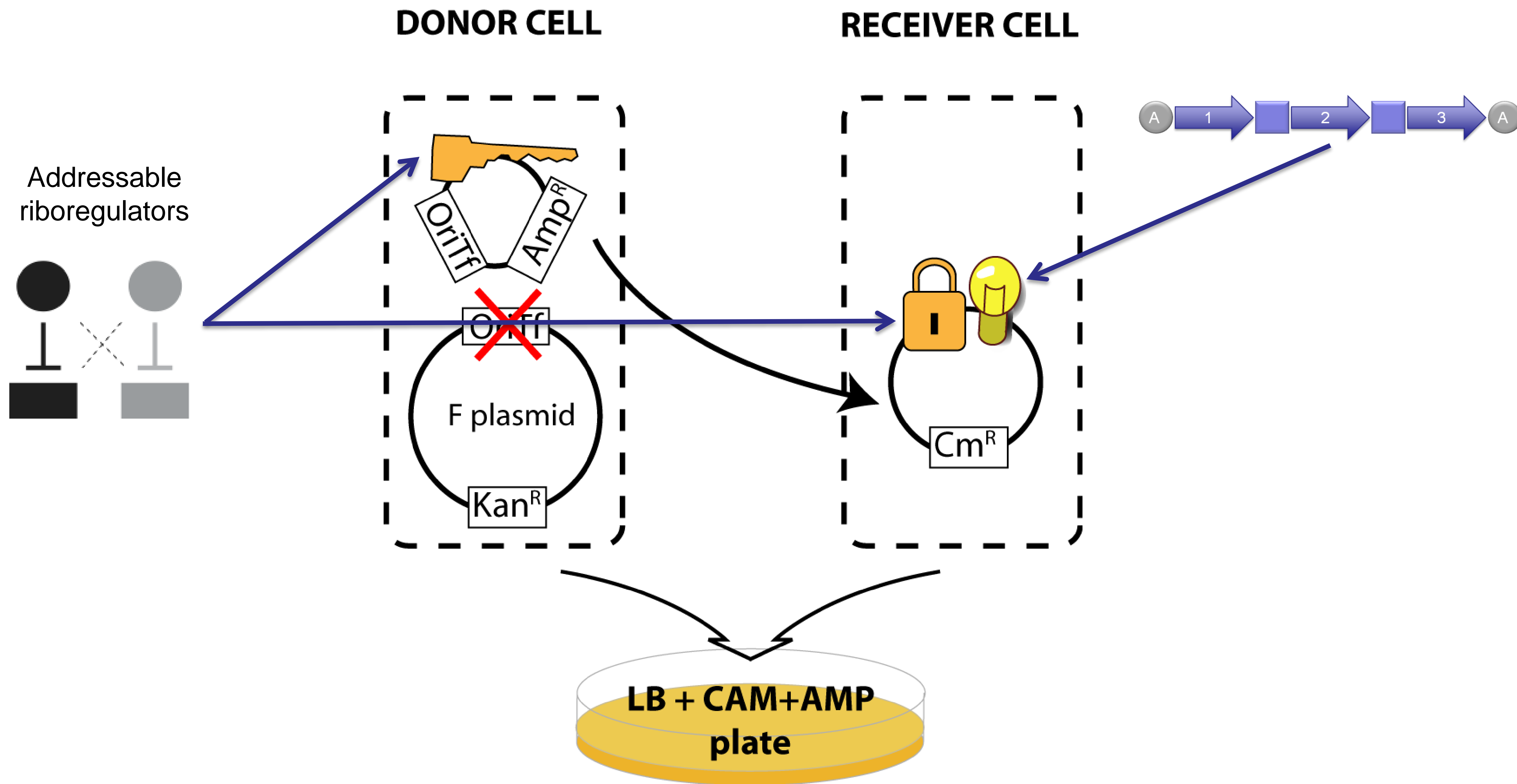
configurations.



If the minimum length of a flippable region is 100 bp and inversion sites are 30bp then the device is less than 2*30*N+100bp long at minimum.

So for N= 10: S~ 10^{11} and Length~ 700 bp

Cell Sends a Coded Message to Recipient Cell



NAND Logic Gates

OriTf

OriTf

F plasmid

KanR

LacI

OriTf

OriTf

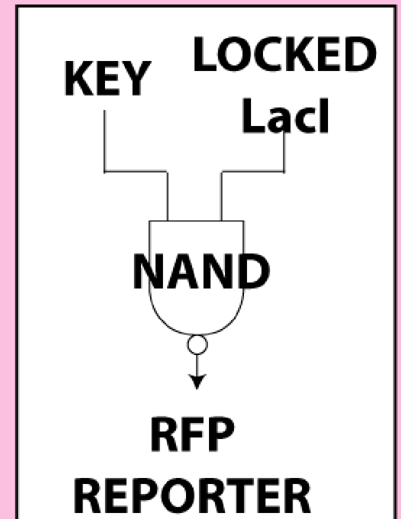
R plasmid

KanR

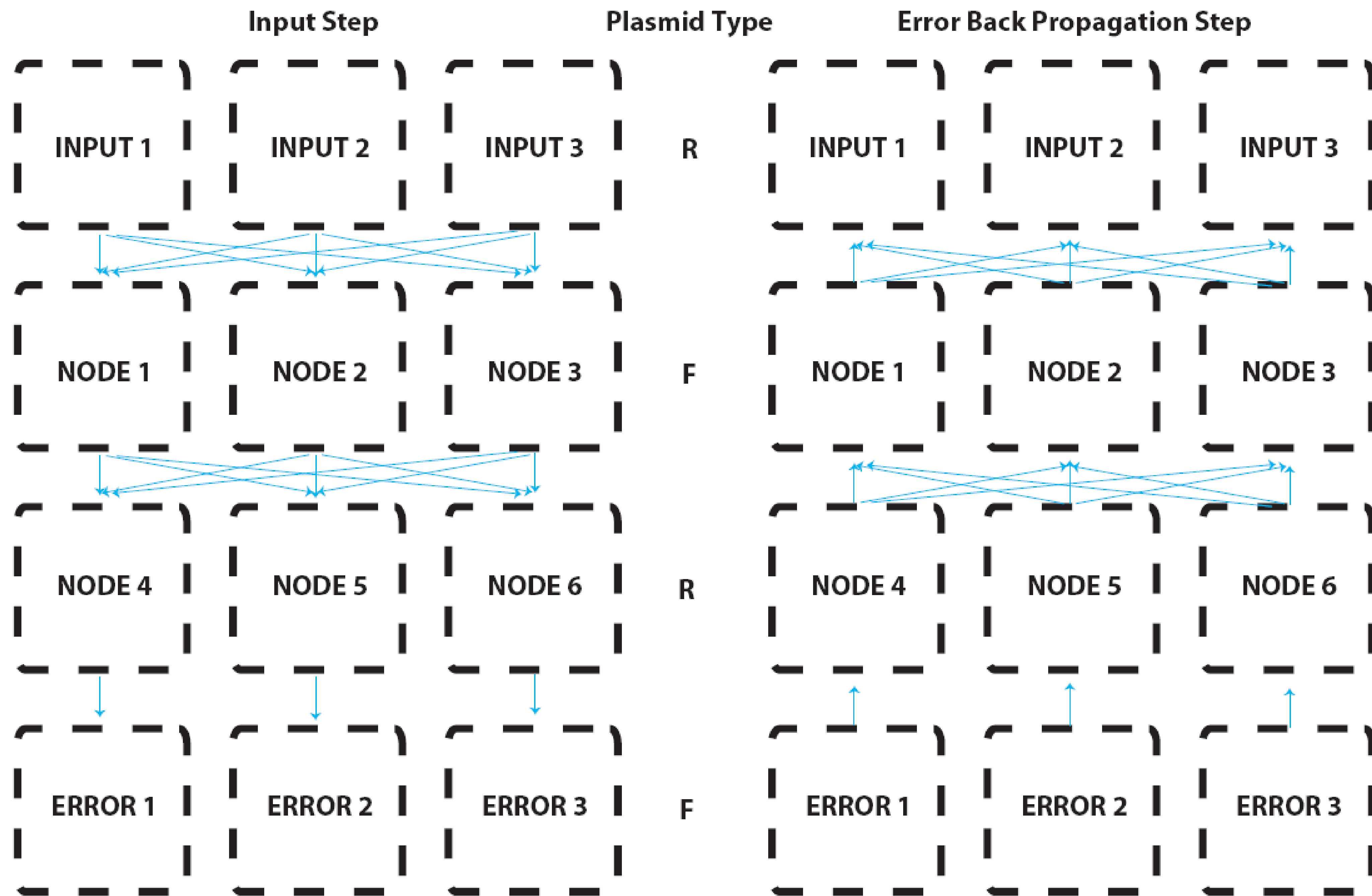
Plac

Truth Table for NAND-gate Behavior

Donor Cell 1 Output	Donor Cell 2 Output	RFP Fluorescence
-	-	+
+	-	+
-	+	+
+	+	-



Trainable Bacterial Networks



Some numbers to contemplate

10^6 microorganisms per gram of soil (10^{33} on earth)

10^9 in a ml of rich media
Divide every 20 minutes
Flip-operations-per-molecule-per-second-per cell ~ 0.01
100-200 plasmids per cell.
5-6 “machines” per plasmid
100-1000 “machines” per genome
Plasmids passed per cell per generation ~ 0.01

What is the computational capacity?

Performance Assessment

General Considerations:

1. Modern computers have advanced to where solving a general P problem does not represent a compelling reason for developing a new technological platform beyond silicon in itself. Thus, solving NP-hard problems represents the main stimulus behind this and other approaches. And within those, NP-complete represents the most immediately interesting class.

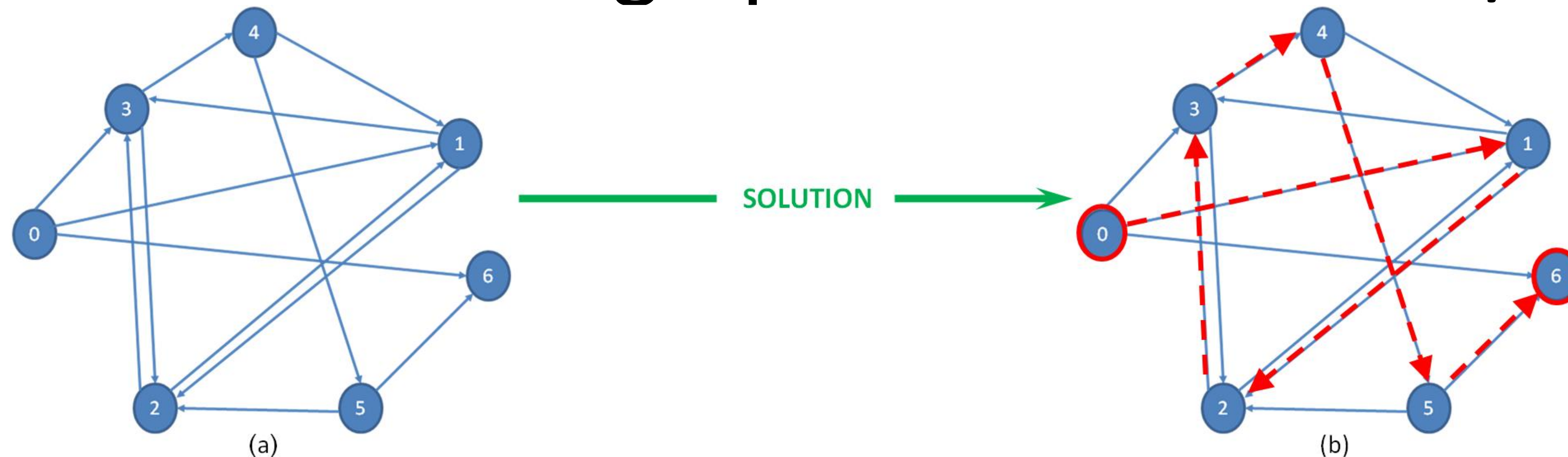
2. At present, NP-complete problems have at best $O(2^N)$ algorithms, i.e. regardless of the implementation – *in silico* or *in situ* – finding a solution involves scanning through most of the solution space. Therefore, the relevant comparison between an *in silico* and *in vivo* or *in vitro* computer is best made within the context of an NP-complete problem solving application and its number of operations executed per unit time.

A motivating example that doesn't even use the full power of the system

1. Develop a synthetic biological platform for efficient generation of N-object permutation pool (N-OPP) uniformly distributed over states
2. Use biomolecular-scale implementation to generate a full realization of N-OPP state space
3. Engineer a sieving scheme to allow problem-specific searches of N-OPP state space by random sampling
4. As an application, engineer a scheme for NP-complete problem – start with Hamiltionan Path Problem (HPP)
5. Design and implement a scalable bioengineered device – potentially creating a practical scalable solver

HPP Example

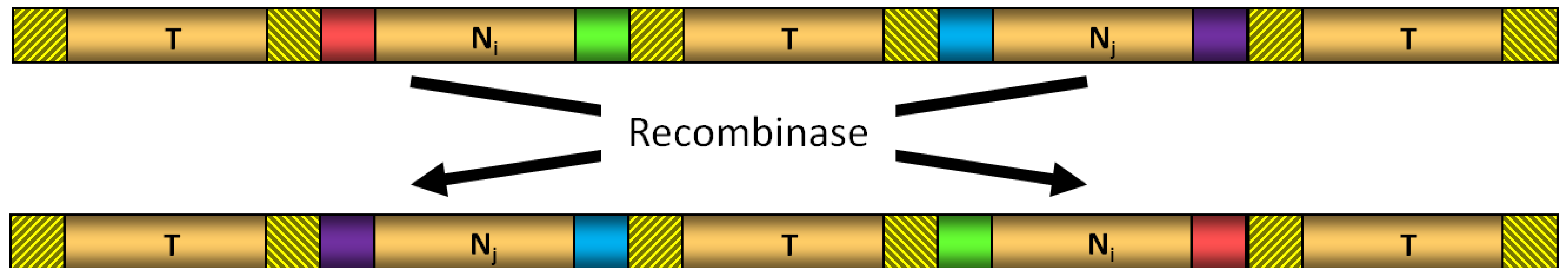
- Problem: find whether a path passing through each node of a graph of size N exists / is valid



- Biomolecular solution:
  1. Generate state space of all paths on a graph that pass through each node only once as N-OPP of N distinct elements on a plasmid (see Slide 2)
  2. Check if any of those paths follow all valid edges (all neighboring node elements in a permutation have corresponding valid edges on the graph), i.e. (…,Ni,Nj,…) = Eij exists on the graph for all (i,j).

N-OPP platform

- Use all the same invertase sites.



- Starting with an initial state, the system stochastically samples configuration space to generate all possible permutations

Generating Uniform N-OPP

- With the stochastic N-OPP generator, the state space is described by a distribution over element permutations

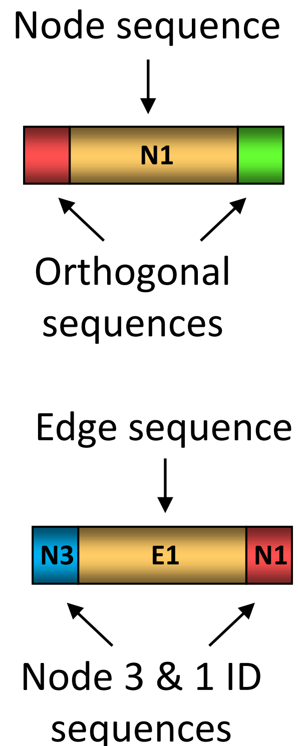
- The desired distribution is ~ uniform, since this insures that no state remains preferentially un-sampled

- With N-OPP represented by sequences of DNA elements on plasmids and recombinase acting on them (Slide 2), the (mixing) time required to reach a distribution exponentially close to uniform is

$$\sim O(N*\mathrm{Log}(N))$$

- (Requires a long and involved derivation)

HPP Sieve – in vitro



Sample first pass implementation:

1. Setup:
   - Design nodes to have unique orthogonal tags at ends
   - Design edges to be oligos flanked by two tags – each complementary to a distinct node
   - Design edge interior to consist of have short homology to the inter-node region with a restriction site T on it
2. Inputs:
   1. A tube w/ plasmids of uniformly distributed N-OPP over nodes
   2. Resuspended oligos of the *complementary* graph (i.e. oligos correspond to *missing* edges).
3. Mix with high oligo excess, melt and re-anneal

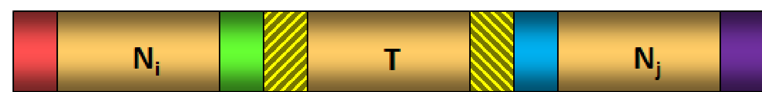
HPP Sieve – in vitro

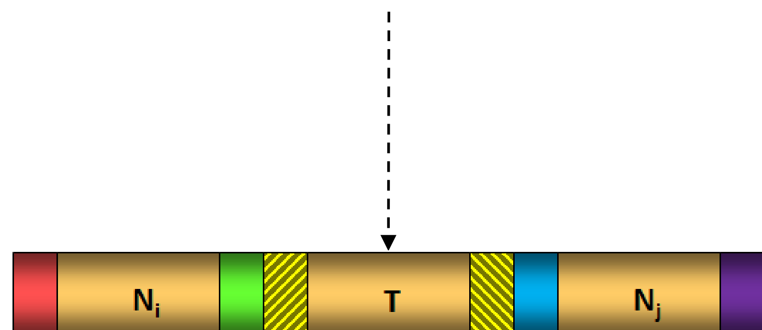
Sample first pass implementation:

4. Sieving correct solution

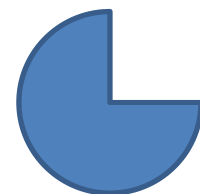
- For any given plasmid containing any given permutation of nodes (a "test path"), if there is an edge complementary to some pair of nodes => this is *not* a valid path (since edges are from the complementary graph and the path is traversing missing edges)
- Binding of an edge oligo to the test path (i.e. the test path is incorrect) generates dsDNA
- Restriction enzyme cuts site T of dsDNA, but not unpaired ssDNA, eliminating only those test paths that bound an oligo and *leaving valid test paths intact*

No complementary edge

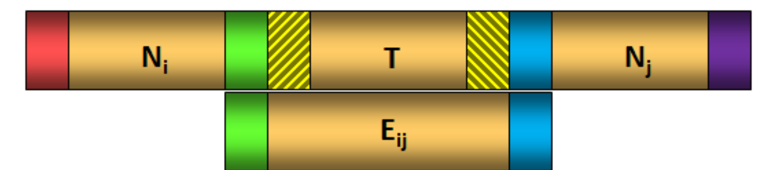


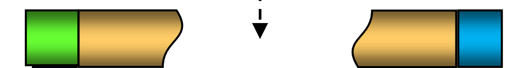


Restriction enzyme



Complementary edge





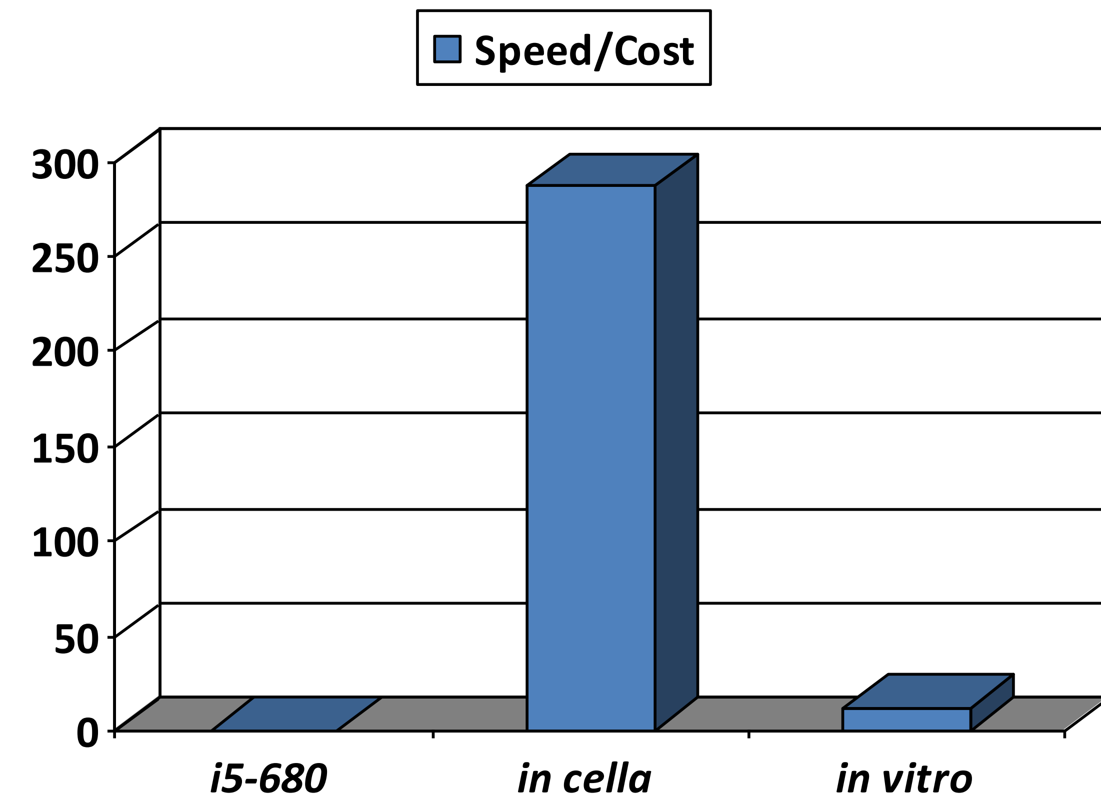
Exo 1

5. Presence of the path can now be detected by PCR

Performance Assessment

<u>Core Statistics</u>: (First-pass implementation)

	i5-680 CPU	***in vivo***	***in vitro***
Speed (GHz/mL)	1.1	29.9	12.2
Power Required	73W	Low	0
Cost ($ per GHz)	~$100	~$0.1	~$1



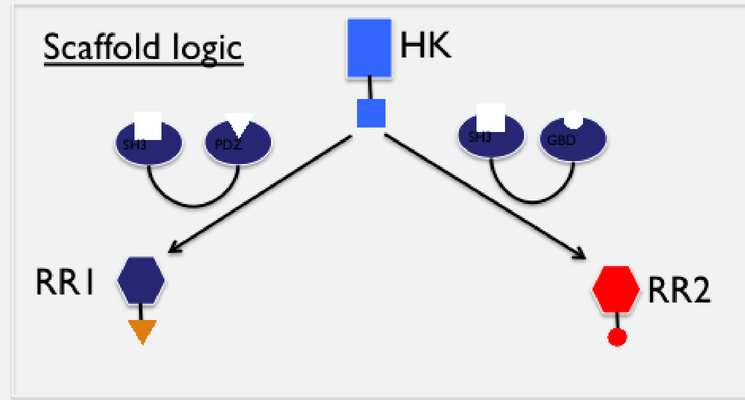
1. Scalability:

   It is essentially infeasible to allocate or maintain > 10,000 CPUs for most computational tasks, but possible to do with > 100 L of culture

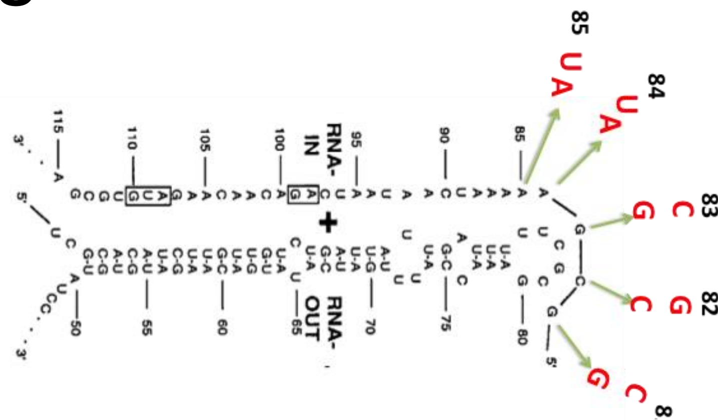
2. These numbers characterize the speed and efficiency of the biomolecular device in generating the N-OPP space, but not the sieving scheme for problem-specific searches

Towards a predictable control layer in cellular engineering

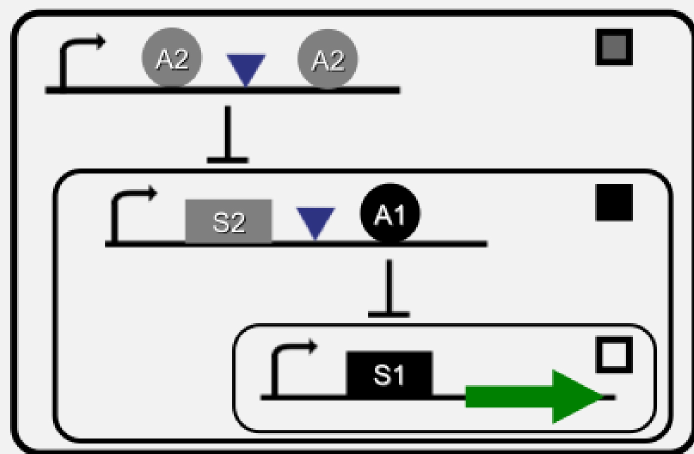
Programmable sensing



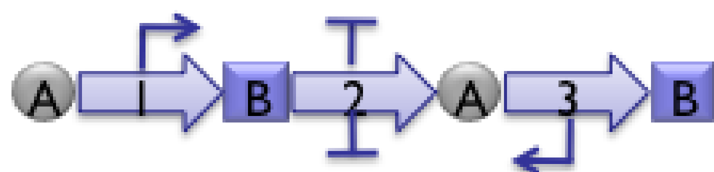
Programmable translation



Programmable transcription



Programmable DNA



Goal: Make the programming the logic of behavioral control "easy"

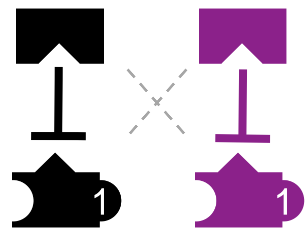
We need systems that are:

Orthogonal
Composable
Connectable
Homogeneous
Designable
Scalable

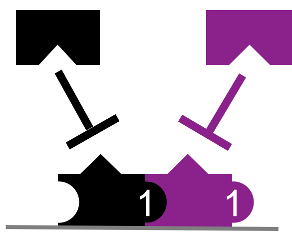
Robust to "Context"

The Larger Picture: Taking Design to the Next Level

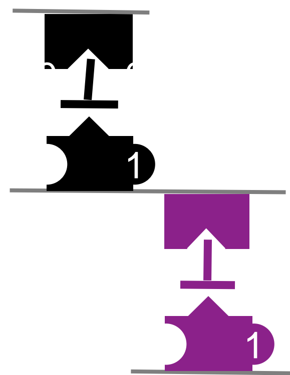
Orthogonality



Composability



Connectivity



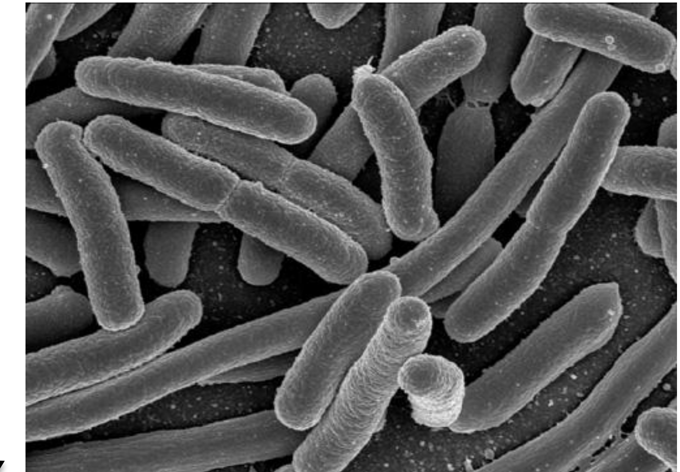
Sense Environmental Signals

Modular Signaling

Process Information and Make Decisions

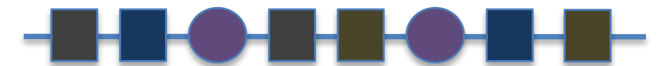
Carry Out Regulatory Programs

S1 S2 A1 GFP WT RFP MT



Rival Complexity of Nature

Persistent state machines



Conclusions

- We are at the edge of creating live organisms for critical human application.
- Therapeutic viruses and bacteria are being built from bottom up and require multiscale design and characterization from molecules to ecologies– ***and requires the building of the proper analytical and computational infrastructure.***
- Using engineering principles is at least creating a network effect in providing parts, devices and systems for other researcher to create new artifacts
  - Chassis is fundamental module for therapeutic bacteria.
- We ARE making progress in developing truly homogeneous designs for scalable circuit engineering in cells- but we are at the proof-of principle phase.
- **We need a robust programming paradigm using our devices as a basis and scaling through population behaviors**.

Join up!

- Synthetic Biology Institute launches in July with our Founding Partner Agilent
- BIOFAB biofab.org

- SYNBERC