

# Formalizing Modular Assembly Systems

Austin Che

March 17, 2003

## 1 Motivation

We would like to assemble biological components in a general and modular way. We abstract an individual module as a string, a DNA sequence. Given any two modules, we would like a general way to combine the two modules such that the combination is itself a usable module that can be combined with other modules.

**Definition** A module  $w$  is a string of characters from a finite set  $\Sigma$ .

A reaction  $\mathcal{R}$  is defined as a set of DNA sequences.

An operation involves adding, removing, or modifying the DNA sequences in a reaction in a specified way.

**Definition** A system  $\mathcal{S}$  is defined by  $\mathcal{S} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$ .

$\mathcal{P}$  is the packaging function that determines the actual sequence that modules come in.  $\mathcal{P}(w) = \{P_0wS_0, P_1f_1(w)S_1, \dots, P_nf_n(w)S_n\}$  for a constant  $n \geq 0$ .  $f_i(w)$  is a sequence that may be dependent on  $w$ . All modules will be packaged and these are the only sequences that can be used in reactions.

$\mathcal{V}$  is the validity function such that  $\mathcal{V}(w)$  is true iff  $w$  is a valid module for the system.

$\mathcal{A}$  is the assembly function such that  $\mathcal{A}(u, v) = w$ .

$\mathcal{M}$  is the biological mechanism for implementing  $\mathcal{A}(u, v) = w$ . Given the sequences  $\mathcal{P}(u)$  and  $\mathcal{P}(v)$ ,  $\mathcal{M}$  produces the sequences  $\mathcal{P}(w)$  using a fixed set of operations on reactions. We require that  $\mathcal{M}$  have the property of *generality* by being independent of the input sequences.

**Definition** System  $\mathcal{S} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$  is *complete* if for any  $u, v$  such that  $\mathcal{V}(u)$  and  $\mathcal{V}(v)$ , then  $\mathcal{V}(\mathcal{A}(u, v))$ ,  $\mathcal{A}(u, v) = xyvz$ , where  $x, y, z$  are fixed strings, and  $\mathcal{M}$  returns the sequences in  $\mathcal{P}(\mathcal{A}(u, v))$ .

System  $\mathcal{S}$  is *sound* if for any valid  $u, v$ ,  $\mathcal{M}$  returns only strings in  $\mathcal{P}(\mathcal{A}(u, v))$ . That is, a system is sound if  $\mathcal{M}$  does not return an incorrect value.

System  $\mathcal{S}$  is *good* if it is complete and sound.

Also, we will only consider non-trivial systems where there are an infinite number of  $w$  for which  $\mathcal{V}(w)$ .

**Definition** Basic operations that are always available include:

*Add*( $\mathcal{R}, w$ ): Adds the sequence  $w$  to reaction  $\mathcal{R}$

*Mix*( $\mathcal{R}_1, \mathcal{R}_2$ ): Adds all sequences in  $\mathcal{R}_2$  into  $\mathcal{R}_1$

## 2 BioBricks

As an example of the above formal definition, we will define the BioBricks assembly system. This has been slightly modified from the original protocol to allow formalization and proof.

**Definition** The BioBricks system  $\mathcal{BB} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$  is defined as follows. Let E, N, X, S, and P represent the EcoRI, NotI, XbaI, SpeI, and PstI restriction recognition sites respectively. M will represent the mixed XbaI/SpeI site.

$$\mathcal{P}(w) = \{ \text{ENX } w \text{ SNP} \} \quad (n = 0)$$

$\mathcal{V}(w)$  is true iff ENX $w$ SNP contains exactly one instance of a substring matching each of E, N, X, S, and P.

$$\mathcal{A}(u, v) = uMv$$

$\mathcal{M}$  exists and will be defined below.

**Remark** For simplicity, this system does not have the extra separator bases around the XbaI/SpeI sites. In addition, the  $\mathcal{V}(w)$  is defined as such so that the mechanism can be proved complete. It is not equivalent to saying that  $w$  does not contain any of the given restriction sites. In particular, it says that  $w$  can not begin with the sequence ATTC that would combine with the GA at the end of the XbaI site to form an extra EcoRI site.

**Definition**  $Cut(\mathcal{R}, enzyme)$ : Finds all occurrences of *enzyme*'s recognition site in sequences in  $\mathcal{R}$  and splits the sequence into two pieces at the appropriate location.

$Purify(\mathcal{R}, length)$ : Remove all sequences of a specific length.

$Ligate(\mathcal{R})$ : Cohesively ligate all sequences that have matching overhangs.

**Lemma 2.1** *BioBricks system  $\mathcal{BB}$  can be implemented with the following biological mechanism  $\mathcal{M}$ .*

**Proof** Given modules  $u$  and  $v$ , we have the sequences ENX $u$ SNP and ENX $v$ SNP. We need to construct module  $w = uMv$  by making ENX $uMv$ SNP.

Let  $F = \text{ENX}u\text{SNP}$  and  $B = \text{ENX}v\text{SNP}$ . Proceed as follows:

1.  $Add(\mathcal{R}_1, F)$
2.  $Cut(\mathcal{R}_1, SpeI)$
3.  $Purify(\mathcal{R}_1, l)$ , where  $l$  is length of the SNP fragment
4.  $Add(\mathcal{R}_2, B)$
5.  $Cut(\mathcal{R}_2, XbaI)$
6.  $Purify(\mathcal{R}_2, m)$ , where  $m$  is length of the ENX fragment
7.  $Mix(\mathcal{R}_1, \mathcal{R}_2)$
8.  $Ligate(\mathcal{R}_1)$

$\mathcal{R}_1$  will contain the desired DNA of ENXuMvSNP ■

**Lemma 2.2** *In  $\mathcal{BB}$ , given any valid modules  $u, v$ ,  $uMv$  is also valid.*

**Proof** To show  $\mathcal{V}(uMv)$ , we need to show that ENXuMvSNP contains only a single E, N, X, S, and P as a substring.  $u$  and  $v$  are valid so if there exists an extra occurrence of a substring matching one of these sites, it must cross either the  $uM$  boundary or the  $Mv$  boundary. The sequence for M is identical to the SpeI site except for the last base. Therefore, if  $uM$  is not valid, then  $uS$  would not be valid, which is a contradiction. Similarly, M is identical to the XbaI site except for the first base. Therefore, if  $Mv$  is not valid, then  $Xv$  is not valid, which is also a contradiction.

Thus,  $uMv$  is valid. ■

**Lemma 2.3** *BioBricks system  $\mathcal{BB}$  is complete.*

**Proof** Given  $u, v$  such that  $\mathcal{V}(u)$  and  $\mathcal{V}(v)$ .

$A(u, v) = w = uMv = xyvz$  where  $y = M$  and  $x = z = \epsilon$ . By the above lemma, as both  $u$  and  $v$  are valid,  $\mathcal{V}(uMv)$ . It is also obvious that  $\mathcal{M}$  works to produce the required sequence.

Thus,  $\mathcal{BB}$  is complete. ■

**Lemma 2.4** *The mechanism  $\mathcal{M}$  for BioBricks system  $\mathcal{BB}$  is sound.*

**Proof** To show soundness of  $\mathcal{M}$ , it is sufficient to show that only one species of DNA will be produced. The first two cuts can only cut in the single expected location as both  $u$  and  $v$  are valid. The ligation will only ligate the XbaI and SpeI site overhangs as no other overhangs have been created, forming the mixed site as expected.

Thus,  $\mathcal{BB}$  is sound. ■

**Theorem 2.5** *The BioBricks system  $\mathcal{BB}$  is good.*

**Proof**  $\mathcal{BB}$  is complete and sound from above lemmas and so  $\mathcal{BB}$  is good. ■

### 3 Recombination System

**Definition** Two strings  $x$  and  $y$  have *homology* if  $x = uAv, y = wAz$ , and  $|A| > \gamma$ , where  $\gamma$  is a fixed constant and  $\gamma > 0$ .

**Definition**  $Rec(\mathcal{R})$ : The recombination operation  $Rec$  adds sequences to reaction  $\mathcal{R}$ . For sequences  $s_i$  and  $s_j$  in  $\mathcal{R}$ , if  $s_i = uAvBw$  and  $s_j = xAyBz$  with  $|A| > \gamma$  and  $|B| > \gamma$ , then the sequences  $uAyBw$  and  $xAvBz$  are added to the reaction if they do not already exist. All possible sequences are added until no new sequences can be added.

**Definition**  $Sel(\mathcal{R}, sequence)$ : The selection operation  $Sel$  removes all sequences from reaction  $\mathcal{R}$  that do not contain the sequence.  $Sel(\mathcal{R}, -sequence)$  removes all sequences from the reaction that do contain the sequence.

**Definition** Let a recombination system  $\mathcal{RE} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$  be defined such that  $\mathcal{M}$  uses only operations *Rec* and *Sel*.

**Definition** Define  $w_f$  to be the first  $\gamma$  base pairs of  $w$  and  $w_l$  to be the last  $\gamma$  base pairs of  $w$ .

**Remark** Without loss of generality, we will assume that all modules are not required to begin or end with a fixed sequence. That is, if modules are all of the form  $AwB$ , then redefine the module to be just  $w$  and let  $A$  and  $B$  fall into the prefix, suffix, and other fixed strings.

We also assume that the  $\mathcal{P}(w)$  does not depend on  $\gamma$ . If this were not the case, a good  $\mathcal{RE}$  system can be shown to exist but would be completely impractical, requiring  $2^\gamma$  sequences for every module.

**Lemma 3.1** *For any general mechanism  $\mathcal{M}$  that uses a fixed series of operations of *Rec* beginning with  $\mathcal{P}(u)$  and  $\mathcal{P}(v)$ , a sequence containing  $u_l Y v_f$  where  $|Y| \leq \gamma$  cannot be generated for an infinite number of valid  $u$  and  $v$ .*

**Proof** First, we show that there exist an infinite number of valid  $u$  and  $v$  for which neither  $\mathcal{P}(u)$  nor  $\mathcal{P}(v)$  contain a sequence with  $u_l Y v_f$ . If this were not the case, then  $\mathcal{P}(w)$  would need to include  $w_l Y x_f$  for every possible  $x_f$ . As we assume that modules do not have a fixed beginning sequence, then the size of the  $\mathcal{P}(w)$  would need to be at least equal to the number of possible  $x_f$  or  $2^\gamma$ , which we have assumed otherwise above.

Now, we show that the general mechanism  $\mathcal{M}$ , using only *Rec*, cannot create a sequence of the form  $u_l Y v_f$  where  $|Y| \leq \gamma$  starting from  $\mathcal{P}(w)$ , a set of sequences which are not of this form. Consider the *Rec* operation needed to construct the first sequence of the form  $u_l Y v_f$  for  $|Y| \leq \gamma$ . Let the input sequences for this operation be  $I_1 = qArBs$  and  $I_2 = xAyBz$ ,  $|A| > \gamma$  and  $|B| > \gamma$ , to form  $I_3 = xArBz$ , which will contain  $u_l Y v_f$ .

Let us examine where  $u_l$  is  $I_3$ . If  $u_l$  is contained in the  $Bz$  part, then  $Bz$  must contain all of  $u_l Y v_f$  and the original input sequence  $I_2$  also contained this sequence, contrary to assumption that this was the first such sequence to be formed. If  $u_l$  is contained in the  $r$  part, then for  $|Y| < \gamma$ , if  $v_f$  is also in the  $r$  part, then  $I_1$  already contained this sequence. Therefore,  $v_f$  must be in the  $z$  part. As  $|B| > \gamma$ ,  $|Y| > \gamma$ . The same argument applies if  $u_l$  is in the  $x$  part. If  $u_l$  is in the  $A$  part, then  $v_f$  must be in  $r$  and  $I_1$  already contained the sequence. Therefore, no such sequence with  $|Y| \leq \gamma$  can be formed.

Therefore, there are an infinite number of valid  $u$  and  $v$  for which  $\mathcal{M}$  will not be able to create a sequence containing  $u_l Y v_f$  where  $|Y| \leq \gamma$ . ■

**Corollary 3.2** *If  $\mathcal{RE}$  is general and complete, the assembly function  $\mathcal{A}(u, v) = xyvz$  must be such that  $|y| > \gamma$ .*

**Proof** This follows directly from the above lemma as  $y$  must be fixed and for an infinite number of  $u$  and  $v$ ,  $|y|$  cannot be less than  $\gamma$ . ■

**Theorem 3.3** *Any system  $\mathcal{RE} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$  that is complete is also unsound.*

**Proof** Let  $\mathcal{RE}$  be a complete system. Let  $w_1, w_2, w_3, w_4$  be valid modules. As  $\mathcal{A}(u, v) = xuyvz$ , let  $w_{123} = \mathcal{A}(\mathcal{A}(w_1, w_2), w_3) = xxw_1yw_2zyw_3z$ . Because the system is complete, we know that  $w_{123}$  is a valid module.

Consider the mechanism for assembling  $w_{123}$  and  $w_4$ . As *Rec* is the only function that can create a new species of DNA, there must be at some point a recombination operation to create a sequence that contains the subsequence  $w_{1234} = xw_{123}yw_4z = xxxw_1yw_2zyw_3zyw_4z$ .

As  $|y| > \gamma$  by the above lemma, the *Rec* operation when adding  $w_{1234}$  to the reaction must also add the sequences that you get by recombining  $w_{1234}$  with itself. For example,  $w'_{1234} = xxxw_1yw_3zyw_2zyw_4z$  would necessarily be created.

$w_{1234}$  and  $w'_{1234}$  are indistinguishable in the system. Consider another case of assembling  $w_{132}$  and  $w_4$  to get  $w_{1324} = w'_{1234}$ . The mechanism must successfully select  $w_{1234}$  in the first case and  $w'_{1234}$  in the second case. This would have to involve doing something different depending on the inputs, which contradicts its generality. Note that the mechanism could use some information stored in the packaging sequences  $\mathcal{P}(w)$ . However, the number of sequences in  $\mathcal{P}(w)$  must be finite, whereas the number of modules that may be assembled in any order is infinite, implying there exists indistinguishable sequences like above.

Therefore, for every sequence containing  $w_{1234}$  that  $\mathcal{M}$  ends with, it must also create the same sequence with  $w'_{1234}$  replacing  $w_{1234}$ , showing that  $\mathcal{RE}$  is unsound. ■

**Corollary 3.4** *No good system exists using only Rec and Sel.*

## 4 Rec/Xis System

There does exist a good system if one adds a *Xis* function.

**Definition**  $Xis(\mathcal{R})$ : The excision operation *Xis* removes fragments from reaction  $\mathcal{R}$ . For all fragments that match  $uAvBw$ , *Xis* replaces the sequence with  $uCw$ , where  $C$  is not equal to either  $A$  or  $B$ .

**Definition** The *Rec/Xis* system  $\mathcal{RX} = (\mathcal{P}, \mathcal{V}, \mathcal{A}, \mathcal{M})$  is defined as follows. Let  $F, E, X$  be fixed arbitrary sequences of length equal to  $\gamma$ .  $P_1$  and  $P_2$  are the coding sequences for positive selection mechanisms (e.g  $P_1 = amp$ ,  $P_2 = kan$ ).  $N_1$  is the coding sequence for a negative selection mechanism (e.g  $N_1 = sacB$ ). Let  $L=attL$  and  $R=attR$ , where  $attL$  and  $attR$  are the sites recognized by phage  $\lambda$ .

$$\mathcal{P}(w) = \{ F w E, FXP_1N_1Rw_f, w_lLP_2XE \} \quad (n = 2)$$

$$\mathcal{A}(u, v) = uBv, \text{ where } B=attB.$$

$\mathcal{V}(w)$  is true iff neither  $wB$  nor  $Bw$  has homology to either  $FXP_1N_1R$  or  $LP_2XE$ . Also  $|w|$  must be greater than  $\gamma$

$\mathcal{M}$  is defined below.

**Remark** The  $\lambda$  *red* system requires overhangs of about 40bp ( $\gamma = 40$ ) and implements the *Rec* function.

The  $\lambda$  *xis*, *int* genes implement the *Xis* function. They excise everything between *attL* and *attR*, replacing it with a new site *attB*. *attB* has the structure *BOB'* and consists of 30bp. Normally,  $\lambda$  integrates from its *attP* site with the structure *attP = POP'*. The *O* is 15bp in length and identical in *attB* and *attP*. *P* is 150bp long and *P'* is 90bp long. Integrating *attP* with *attB* forms *attL = BOP'* and *attR = POB'*.

**Lemma 4.1** *The system  $\mathcal{R}\mathcal{X}$  can be implemented with the following biological mechanism  $\mathcal{M}$ .*

**Proof** The following mechanism is a scheme using biological components from bacteriophage  $\lambda$ . Cells are assumed to contain the  $\lambda$  *int*, *xis* genes and the *red* genes under separate inducible control.

We have as input the sequences  $\mathcal{P}(u) = \{ F u E, FXP_1N_1Ru_f, u_lLP_2XE \}$  and  $\mathcal{P}(v) = \{ F v E, FXP_1N_1Rv_f, v_lLP_2XE \}$  and we need to construct the sequences  $\mathcal{P}(w) = \mathcal{P}(uBv) = \{ F w E, FXP_1N_1Rw_f, w_lLP_2XE \}$ .

1. *Add*( $\mathcal{R}_1, FuE$ ) and *Add*( $\mathcal{R}_1, u_lLP_2XE$ )
2. *Rec*( $\mathcal{R}_1$ )
3. *Sel*( $\mathcal{R}_1, P_2$ ) to obtain  $I_1 = FuLP_2XE$ .
4. *Add*( $\mathcal{R}_2, FvE$ ) and *Add*( $\mathcal{R}_2, FXP_1N_1Rv_f$ )
5. *Rec*( $\mathcal{R}_2$ )
6. *Sel*( $\mathcal{R}_2, P_1$ ) to obtain  $I_2 = FXP_1N_1RvE$ .
7. *Mix*( $\mathcal{R}_1, \mathcal{R}_2$ )
8. *Rec*( $\mathcal{R}_1$ )
9. *Sel*( $\mathcal{R}_1, P_1$ ) and *Sel*( $\mathcal{R}_1, P_2$ ) to obtain the sequence  $I_3 = FuLP_2XP_1N_1RvE$
10. *Xis*( $\mathcal{R}_1$ )
11. *Sel*( $\mathcal{R}_1, -N_1$ ) to obtain the sequence  $I_4 = FuBvE = FwE$ . This is the first sequence for  $\mathcal{P}(uBv)$ . Use  $FXP_1N_1Ru_f$  and  $v_lLP_2XE$  for the other two sequences as  $w_f = u_f$  and  $w_l = v_l$ .

Thus,  $\mathcal{R}\mathcal{X}$  is biologically implementable. ■

**Proposition 4.2** *System  $\mathcal{R}\mathcal{X}$  is complete.*

**Proof** Given  $u, v$  such that  $\mathcal{V}(u)$  and  $\mathcal{V}(v)$ .

$A(u, v) = w = uBv = xyvz$  where  $y = M$  and  $x = z = \epsilon$ . As both  $u$  and  $v$  are valid, neither  $uB$  nor  $Bv$  contain an invalid substring. The only remaining case for an invalid homology match is for a substring to begin in  $u$ , contain  $B$ , and end in  $v$ . Given the appropriate choices for the sequences  $F, E, X, P_1, P_2, N_1$ , they can be chosen so they do not contain  $B$  as a subsequence. It is believed that attL and attR do not contain  $attB$  as a subsequence. If this is not the case, then the system can be made complete by disallowing modules that begin or end with sequences that could possibly create a sequence that matches any of the invalid substrings.

Thus,  $\mathcal{R}\mathcal{X}$  is complete. ■

**Proposition 4.3** *The system  $\mathcal{R}\mathcal{X}$ , under reasonable conditions, is sound.*

**Proof** The mechanism  $\mathcal{M}$  is believed to be biologically and theoretically sound, assuming that modules  $u$  and  $v$  do not have homology. Other possible recombinations are assumed to not occur if the selections work efficiently. In particular, the  $attB$  30bp sequence is presumed to not be long enough for recombination to occur.

**Proposition 4.4**  *$\mathcal{R}\mathcal{X}$  is good under the conditions that only modules that have no homology to each other are assembled together.*

**Proposition 4.5** *No Rec/Xis system exists with  $n = 0$*