MCB Toolbox

Tianyi Shi

2020-01-23

Contents

For the second tutorial, I want you to try some proper molecular biology using online tools, just as we do every day in the lab. Try it out on your own first, though you may wish to discuss any issues you may be having with each other. Please bring files/printouts and if you have one, your laptop to the tutorial. You will find NCBI BLAST useful and various links within www.expasy.org (e.g. In proteomics section, Clustal, T-coffee, Translate, prot param etc). You may also like to download ApE or Snapgene viewer from the web but make sure you scan with e.g. Sophos before you open them. You can do full in silico cloning in ApE (but only in the expensive paid-for version of Snapgene) but Snapgene (even free viewer) offers much better visualization so you may want to use both but for different features. For structural productions, Swissprot is convenient, then for mutagenesis, move to Pymol or equivalent.

```
Mystery gene (lower case = non-coding, upper case = coding)
```

```
GTATTTTGCGTCCATTTCGATCAAAAGACTATTTCTCATTTCCTCCCACGATTTTAGATAATAATATTATCTTATGTCATGCACAGATGATAAGTGATGATGA
```

```
pIVEX2.3d Vector Sequence
```

```
TCGCGCGTTTCGGTGATGACGGTGAAAACCTCTGACACATGCAGCTCCCGGAGACGGTCACAGCTTGTCTGTAAGCGGATGCCGGGAGCAGACAAGCCCGTCA
```

1. Use the information in the attached DNA file (mystery gene) to identify the encoded protein and the species.
    - Caenorhabditis elegans Probable Werner syndrome ATP-dependent helicase homolog 1 (wrn-1) (UniProt accession code: Q19546)
2. Draw a schematic diagram of the protein, noting the number of amino acids, predicted Mw and pI of the encoded protein, and annotate on functional domains (e.g. in ppt).
    - 1056 amino acids; MW = 118522.73 Da (average mass)/118447.63 Da (monoisotopic mass); pI = 6.08 (predicted using ExPASy 'Compute pI/Mw tool')
    - Automated homology modelling in the SWISS-MODEL Repository uses 1oyy.1.A, 4cdg.1.A and 4cgz.1.A as templates to obtain models with global QMEANs of -3.35, -3.57 and -4.19, respectively
    - Manual (interactive) modelling using 4q47.1.A (DrRecQ catalytic core (Dr = *Deinococcus radiodurans*)) achieved a better QMEAN of -3.09. The structure is shown in Fig. 1 (highlighting functional domains) and in Fig. 2 (showing local quality (QMEAN) scores). The annotations on functional domains are based on Chen et al. (2014).
    - Update on 2 Feb: yet a higher QEMAN of -2.90 is achieved with 2wwy.1. 2wwy.1 is the structure of human RECQ-like helicase in complex with DNA substrate(Pike et al. 2015). The homology model based on this is shown in Fig. 3, and the PyMOL Session (`.pse`) file can be downloaded here.
    - As shown in Fig. 4, WRN is unique in the family of RecQ helicases in possessing an 3'-to-5' exonuclease domain(Cox and Mason 2013). In *C. elegans*, the exonuclease and the helicase are encoded in separate genes. The two experimental structures used for modelling (4q47.1.A

(DrRecQ) and 2wwy.1 (human RECQ1)) and the query *C. elegans* wrn homolog all represent only the helicase domain and the RQC (WH) domain.

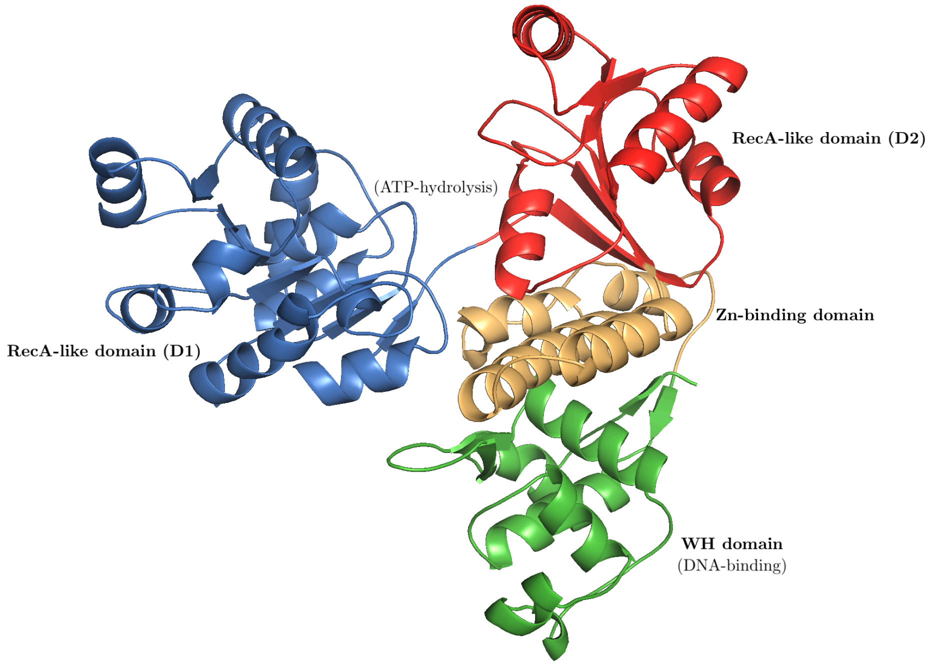


Figure 1: A homology model of *C. elegans* wrn-1 based on DrRecQ catalytic core (PDB accession code: 4Q47). The N-terminal part contains the common core structure of the RecQ helicase family, which comprises two RecA-like domains, D1 and D2, each composed of a central -sheet surrounded by -helices. The conserved sequence motifs at the interface of D1 and D2 are responsible for ATP binding and hydrolysis. The following Zn-binding domain stabilizes the protein structure. The C-terminus consists of a WH domain (a.k.a RQC domain), which directly participates in DNA binding and base pair separation. [@Chen-2014]

3. Find homologues in human, Xenopus and mice and give the protein names (Uniprot should help)
    - Use UniProt BLAST
    - Query: Q19546 (UniProt ID)
    - Relevant Results:
        - Werner syndrome ATP-dependent helicase (Homo sapiens) (Q14191; WRN_HUMAN)
        - Werner syndrome ATP-dependent helicase homolog (Mus musculus (O09053; WRN_MOUSE)
        - Werner syndrome ATP-dependent helicase homolog (Xenopus laevis) (O93530; WRN_XENLA)
4. Show homology between your mystery protein and the human, mouse and frog homologues.
    - Sequences aligned using UniProt alignment tool (accessible via the BLAST result page)
    - results displayed using SnapGene Viewer (view PDF)
    - UniProt uses the 'clustalo' program (MSA algorithm) to align
    - I also used the 'muscle' program (MUltiple Sequence Comparison by Log-Expectation) in the command line to produce a different result, and standardised its output using BioPython for visualisation in SnapGene (view PDF)

```
$ cat ce.fasta hs.fasta mm.fasta xl.fasta | muscle -out muscle.fasta
```

```
from Bio import AlignIO
muscle = AlignIO.read('resources/wrn/muscle.fasta', 'fasta')
AlignIO.write(muscle, 'muscle.aln', 'clustal')
```

With help of the `seqin` R package, the pairwise genetic differences between these 4 protein sequences

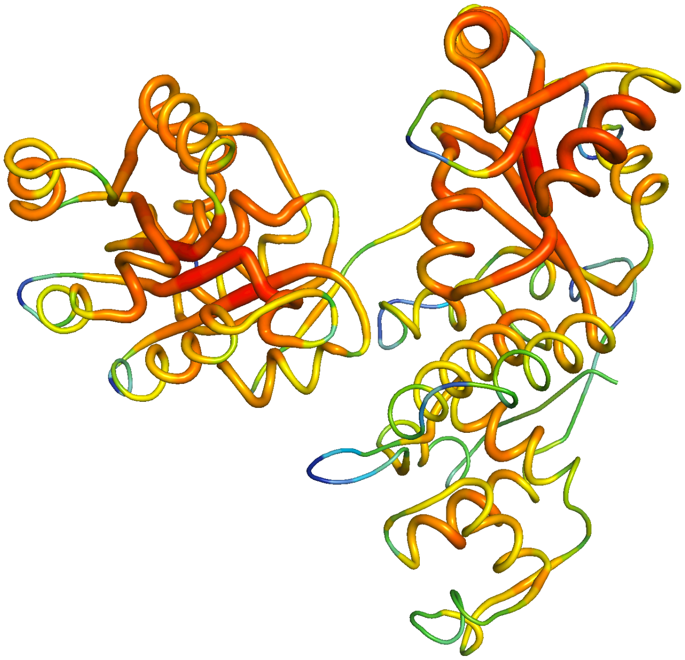


Figure 2: The same homology model, showing local QMEAN scores (b-factor). Compared to green and thin regions, red and thick regions have local geometrical properties deviating further from those of experimentally determined structures. The Zn-binding domain and WH domain are more consistent with experimental structures than the two RecA-like domains.

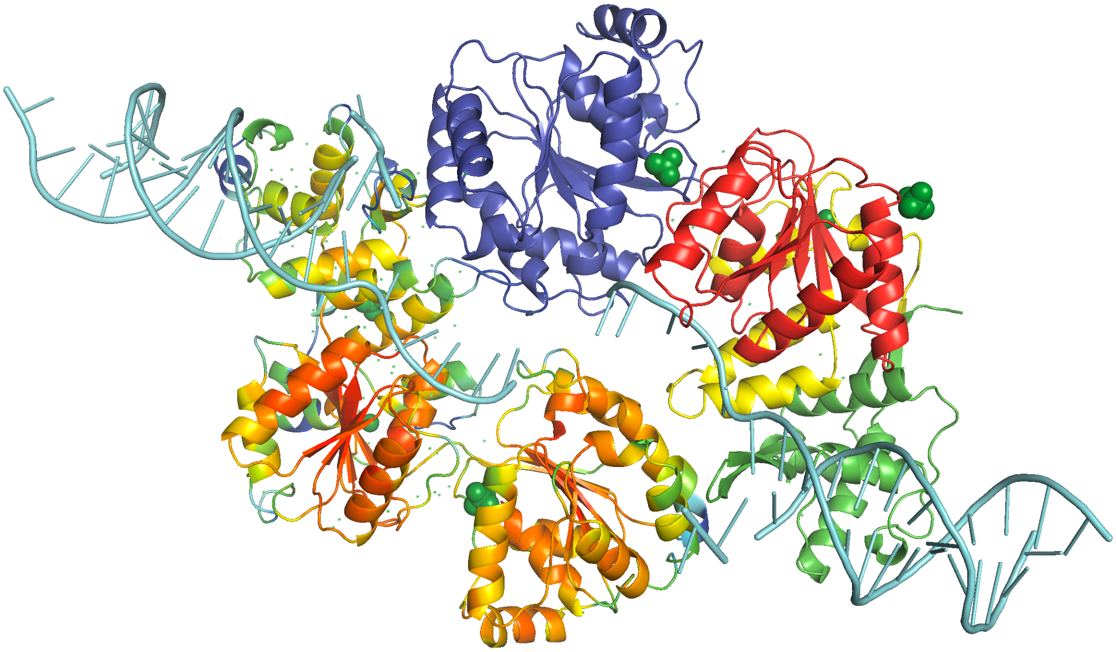


Figure 3: WRN catalytic core (RECQ) in action. This structure is based on the structure of a dimer of human RECQ1 helicase in complex with tailed duplux DNA solved by @Pike-2015. The monomer on the bottom shows the b-factor (local QMEAN) of the homology model with C. elegans wrn-1, where red indicates high and green indicates low scores. The monomer on the top shows the subunits of WRN catalytic core (RECQ), where D1, D2, Zn-binding, and WH (RQC) domains are coloured blue, red, yellow and green, respectively, similar to Fig. 1. DNA is coloured cyan. Ligands (2 zinc ions, 3 sulfate ions and 2 ethanediol molecules) are not removed.

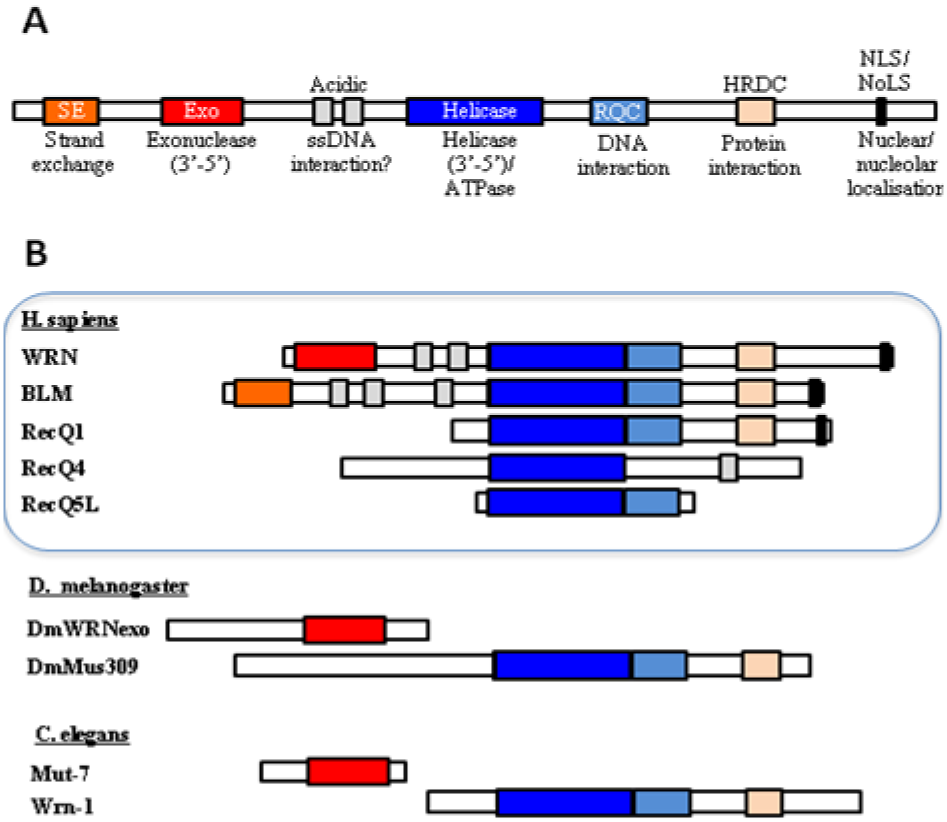


Figure 4: Domain organisation of human WRN and its homologs in *D. melanogaster* and *C. elegans*. Adapted from @Cox-2013.

can be calculated:

```
aln <- seqinr::read.alignment('./resources/wrn/clustal.aln', 'clustal')
alnDist <- seqinr::dist.alignment(aln)
alnDist
```

```
##                       SP|Q19546|WRN_CAEEL SP|Q14191|WRN_HUMAN SP|O09053|WRN_MOUSE
## SP|Q14191|WRN_HUMAN           0.5710006                                        
## SP|O09053|WRN_MOUSE           0.5782777           0.3408993                    
## SP|O93530|WRN_XENLA           0.5903942           0.4396192           0.4499939
```

These can be visulised by a phylogenetic tree produced by the `ape` package, as shown in Fig. 5.

```
mydist <- seqinr::dist.alignment(aln)
mytree <- ape::nj(mydist)
myrootedtree <- root(mytree, 'SP|Q19546|WRN_CAEEL', resolve.root=TRUE)
ape::plot.phylo(myrootedtree, type="p")
```

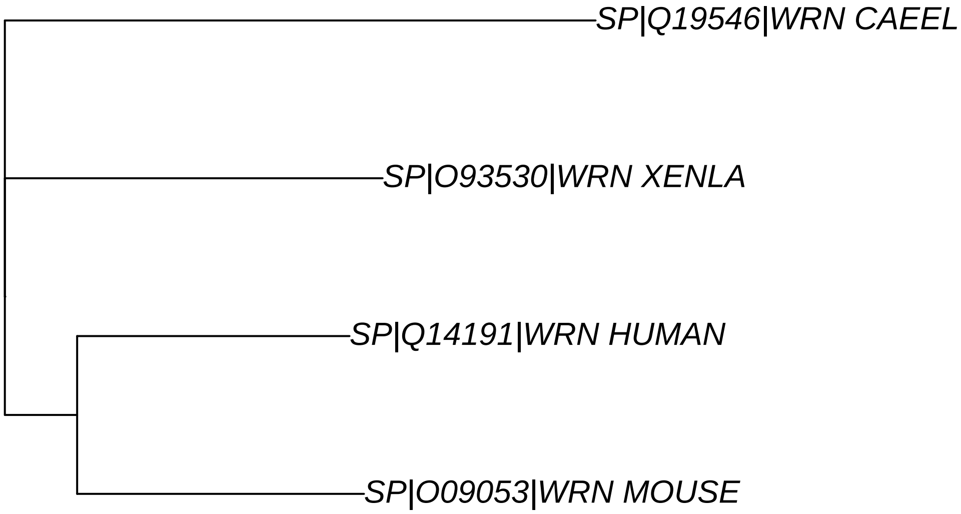


Figure 5: The phylogenetic relationship between the WRN homologs in 4 organisms

5. Design PCR primers to amplify the coding region, incorporating suitable restriction enzyme sites in each primer so that you can clone easily into the vector pIVEX2.3d (though there are many online primer design sites, I find this is best done manually to get a feel for it, but these days I use SnapGene (viewer version is free….). Hint: we want to keep the start codon ATG so chose an enzyme that contains ATG in its recognition site for the forward primer.
    - I opened the original sequence with SnapGene. In the 'Sequence' view, I displayed translated amino acids with 'Frame +1 only' and thus located the stop codon (the start codon is obviously +1)[1]. Then I used the 'Enzymes' view to obtain a list of noncutters. By comparing this list with the MCS of the piVEX2.3d, I selected *Nde*I and *Sal*I as the RE sites to be used[2]. The *Sal*I site is not in frame with the C-terminal His tag and stop codon, so I included an additional random nucleotide 3' to the RE site. The stop codon is excluded from the reverse primer because it is provided in the vector.
    - forward: `NN CAT ATGATAAGTGATGATGACGATCTACC`
        - `CATATG` is *Nde*I restriction site. It ends with `ATG`, so adding `CAT` instead of `CATATG` to the 5'

[1] Why does this sequence start with ATG but not end with a stop codon? Are the additional downstream nucleotides for the convinience of primer design (is it worthwhile to include additional upstream nucleotides)?

[2] Is there a more automatic way of choosing RE sites? I wish I could select the MCS sequence of the vector and the GOI sequence, and then the intersection between the cutting sites of the former and the noncutting sites of the latter is computed automatically. Is this functionality hidden somewhere in SnapGene, or is it available elsewhere? If not I'll try to make it! (A prototype is shown in Fig. 7 and 8)

suffices.
    - 31-mer; 39% GC (>40% if `NN` = `GC`); 26 annealed bases; T_m = 56°C[3]
  - reverse: `NN GTCGAC N AAGTTTGAATTTCTTCAATGGAGGAGC`
    - `GTCGAC` is *Sal*I restriction site
    - 36-mer; 43% GC; 28 annealed bases; T_m = 58°C
  - *In silico* PCR is done in SnapGene, and the resulting DNA can be downloaded here

6. *In silico* clone this into the vector pIVEX2.3d (file attached - .dna is a snap gene format, docx is of course in word). Why might we want to use this vector?
   - This vector has a C-terminal 6-His tag which allows efficient purification by affinity chromatography
   - The MCS is associated with the T7 promoter. T7 RNAP is highly processive, so consecutive expression is expected.
7. Check that your new construct allows correct protein expression in frame with the hexahistidine tag of the vector (SnapGene is good for this, but you can also do it in a few more steps using ApE plus translation apps via Expasy)
   - Yes. (view result as PDF or .dna file)
   - Serendipitously, the MCS has an ATG in frame with our insert and the His-tag/STOP codon, so translation will probably start more upstream than we expected, which will add 3 additional amino acid residues to the N-terminus of the product. This ATG was not in frame with the His-tag before insertion, and thus unspotted by SnapGene or me.[4]

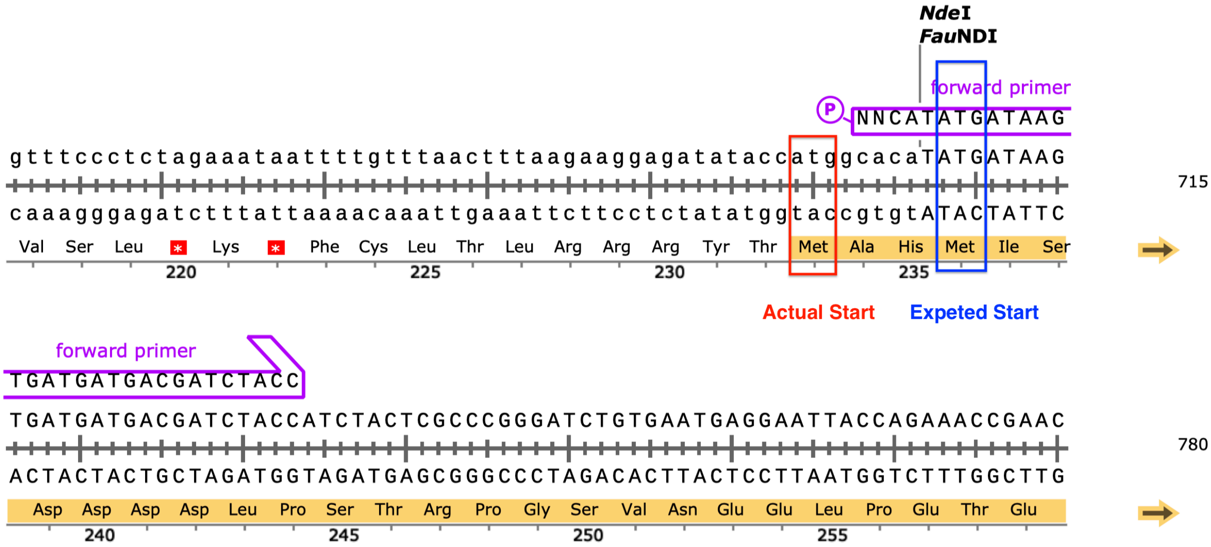

NdeI
FauNDI
forward primer
P
NNCATATGATAAG
gtttccctctagaaataattttgtttaactttaagaaggagatataccatggcacaTATGATAAG
715
caaagggagatctttattaaaacaaattgaaattcttcctctatatggtaccgtgtATACTATTC
Val Ser Leu * Lys * Phe Cys Leu Thr Leu Arg Arg Arg Tyr Thr Met Ala His Met Ile Ser
220
225
230
235
Actual Start
Expeted Start
forward primer
TGATGATGACGATCTACC
TGATGATGACGATCTACCATCTACTCGCCCGGGATCTGTGAATGAGGAATTACCAGAAACCGAAC
780
ACTACTACTGCTAGATGGTAGATGAGCGGGCCCTAGACACTTACTCCTTAATGGTCTTTGGCTTG
Asp Asp Asp Asp Leu Pro Ser Thr Arg Pro Gly Ser Val Asn Glu Glu Leu Pro Glu Thr Glu
240
245
250
255

Figure 6: An additional in-frame start codon (red) upstream to the start codon of the insert (blue).

8. Predict the protein fold of the new protein (eg Swissprot). Show the effect of predicted enzyme active site mutations on the structure (eg new file in swissport, but Pymol is good for this).
   - Swiss-Model shown in Q2.

[3] When calculating %GC and T_m, do we need to include the 5' additional bases used for RE cutting/frame adjustment? In SnapGene it seems that these additional bases are included in the calculation of %GC but not T_m. In practice, isn't the annealing region the complete length of the primers for most DNA strands after a few rounds of synthesis? Do we need to increase the annealing temperature then?

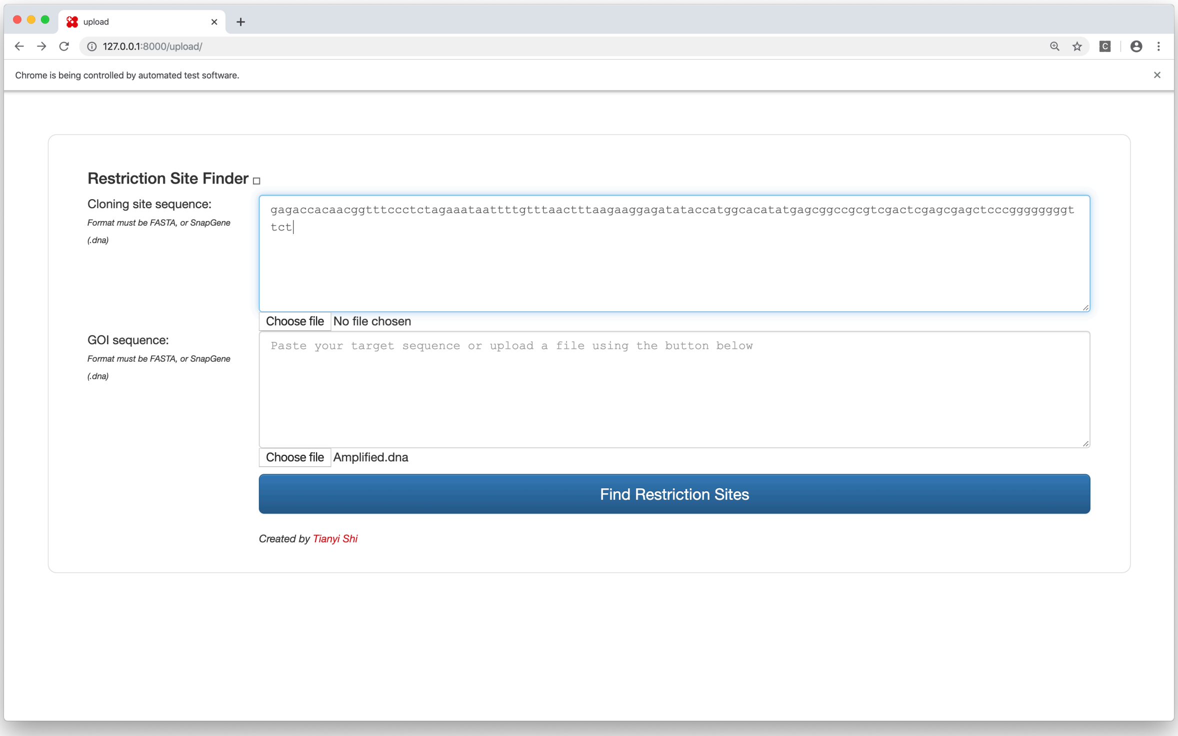
[4] It would be nice if it can highlight all ATGs regardless of the frame. I think additional in-frame ATGs can cause inclusion of unwanted aa residues and addtional out-of-frame ATGs can ruin the expression of GOI, especially in eukaryotes, because their ribosomes start translation at the first encountered ATG. It this correct?

0.0.1 Bonus: Tianyi's Restriction Site Finder

See Fig. 7 and 8.

This tool takes in two sequences (plain text or FASTA or SnapGene `.dna` file), one representing the cloning site and the other representing the GOI, and finds the restriction sites inside the cloning site but not in the GOI, and label them with their positions in the cloning site and their commecial suppliers.

The web interface is built upon the Django framework and the data come from the `Restriction` module of BioPython, which is (sort of) a wrapper around NEB's REBASE.



Chrome is being controlled by automated test software.

Restriction Site Finder □

Cloning site sequence:
Format must be FASTA, or SnapGene (.dna)

```
gagaccacaacggtttccctctagaaataattttgtttaactttaagaaggagatataccatggcacatatgagcggccgcgtcgactcgagcgagctcccggggggggt
tct
```

Choose file No file chosen

GOI sequence:
Format must be FASTA, or SnapGene (.dna)

Paste your target sequence or upload a file using the button below

Choose file Amplified.dna

Created by Tianyi Shi

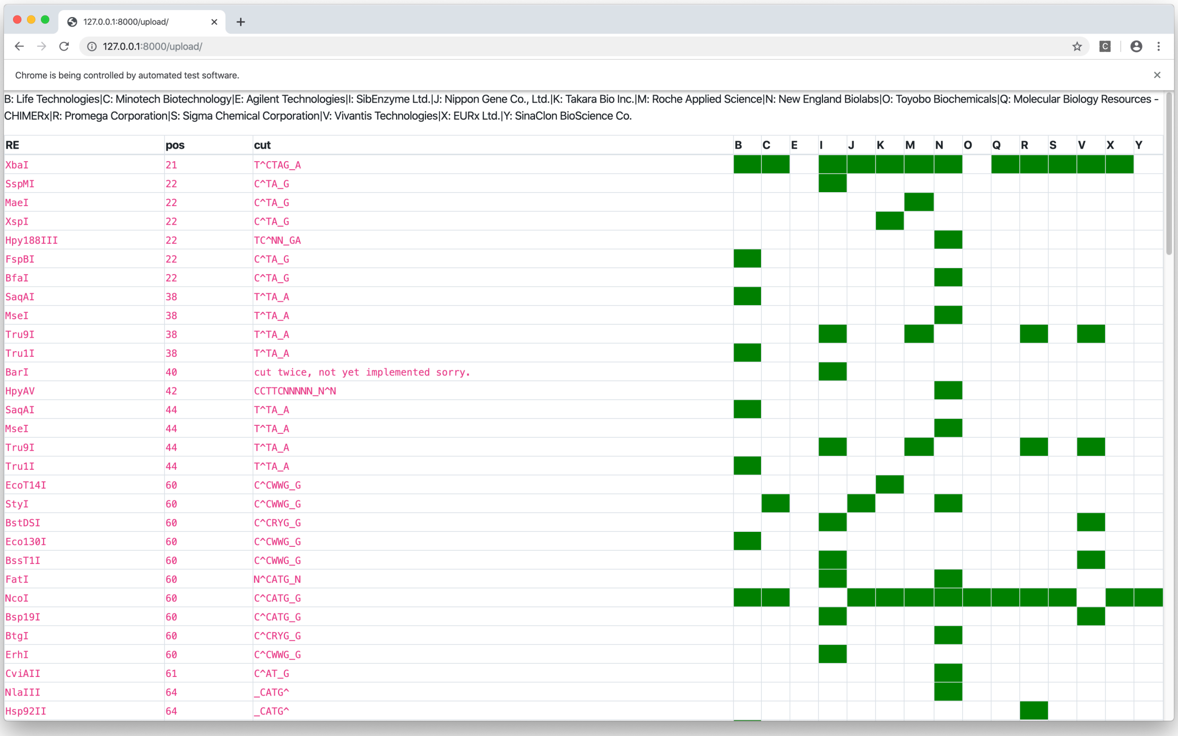
Figure 7: Restriction site finder by Tianyi. Query page.

References

Chen, Sheng-Chia, Chi-Hung Huang, Chia Shin Yang, Tzong-Der Way, Ming-Chung Chang, and Yeh Chen. 2014. "Crystal Structure of Deinococcus Radiodurans Recq Helicase Catalytic Core Domain: The Interdomain Flexibility." *BioMed Research International* 2014. Hindawi.

Cox, Lynne S., and Penelope A. Mason. 2013. "The Role of Wrn Helicase/Exonuclease in Dna Replication." In *The Mechanisms of Dna Replication*, edited by David Stuart. Rijeka: IntechOpen. https://doi.org/10.5772/51520.

Pike, Ashley C. W., Shivasankari Gomathinayagam, Paolo Swuec, Matteo Berti, Ying Zhang, Christina Schnecke, Francesca Marino, et al. 2015. "Human Recq1 Helicase-Driven Dna Unwinding, Annealing, and Branch Migration: Insights from Dna Complex Structures." *Proceedings of the National Academy of Sciences* 112 (14). National Academy of Sciences: 4286–91. https://doi.org/10.1073/pnas.1417594112.



B: Life Technologies|C: Minotech Biotechnology|E: Agilent Technologies|I: SibEnzyme Ltd.|J: Nippon Gene Co., Ltd.|K: Takara Bio Inc.|M: Roche Applied Science|N: New England Biolabs|O: Toyobo Biochemicals|Q: Molecular Biology Resources - CHIMERx|R: Promega Corporation|S: Sigma Chemical Corporation|V: Vivantis Technologies|X: EURx Ltd.|Y: SinaClon BioScience Co.

RE	pos	cut	B	C	E	I	J	K	M	N	O	Q	R	S	V	X	Y
XbaI	21	T^CTAG_A	🟩	🟩		🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟩	🟩	🟩	
SspMI	22	C^TA_G				🟩											
MaeI	22	C^TA_G							🟩								
XspI	22	C^TA_G						🟩									
Hpy188III	22	TC^NN_GA								🟩							
FspBI	22	C^TA_G	🟩														
BfaI	22	C^TA_G								🟩							
SaqAI	38	T^TA_A	🟩														
MseI	38	T^TA_A								🟩							
Tru9I	38	T^TA_A				🟩			🟩				🟩		🟩		
Tru1I	38	T^TA_A	🟩														
BarI	40	cut twice, not yet implemented sorry.				🟩											
HpyAV	42	CCTTCNNNNN_N^N								🟩							
SaqAI	44	T^TA_A	🟩														
MseI	44	T^TA_A								🟩							
Tru9I	44	T^TA_A				🟩			🟩				🟩		🟩		
Tru1I	44	T^TA_A	🟩														
EcoT14I	60	C^CWWG_G						🟩									
StyI	60	C^CWWG_G		🟩			🟩			🟩							
BstDSI	60	C^CRYG_G				🟩									🟩		
Eco130I	60	C^CWWG_G	🟩														
BssT1I	60	C^CWWG_G				🟩									🟩		
FatI	60	N^CATG_N				🟩				🟩							
NcoI	60	C^CATG_G	🟩	🟩			🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩		🟩	🟩
Bsp19I	60	C^CATG_G				🟩									🟩		
BtgI	60	C^CRYG_G								🟩							
ErhI	60	C^CWWG_G				🟩											
CviAII	61	C^AT_G								🟩							
NlaIII	64	_CATG^								🟩							
Hsp92II	64	_CATG^											🟩				

Figure 8: Restriction site finder by Tianyi. Result page.