Design Group-Specific FISH Probes

Erik S. Wright
University of Wisconsin
Madison, WI

October 14, 2013

Contents

1 Introduction **1**

2 The Objective of FISH Probe Design **2**

3 Getting Started **2**
3.1 Installing OligoArrayAux 2
3.2 Startup 2
3.3 Creating a Sequence Database 3
3.4 Defining Groups 3

4 Probe Design Steps **3**
4.1 Tiling Sequences 3
4.2 Designing All Possible Probes 4

5 Dual Probe Design Steps **5**
5.1 Designing the Optimal Dual Probe Set 5
5.2 Visualizing the Target Sites 9
5.3 Finishing Up 10

6 Session Information **10**

1 Introduction

This document describes how to design group-specific FISH probes using the DECIPHER package through the use of the `DesignProbes` function. Fluorescent In-Situ Hybridization (FISH) is an laboratory technique that enables visualizing specific microorganisms in their natural environment using fluorescent microscopy. As a case study, this tutorial focuses on the small-subunit ribosomal RNA (SSU rRNA) from a collection of bacteria found in drinking water. Here we describe how to design probes targeting a group of interest that will not cross-hybridize with other bacteria from the same sample. A similar strategy could be used with any set of aligned sequences that are separated into groups. For example, genus-specific probes designed with this program that target all named bacteria and archaea genera are provided online at `http://DECIPHER.cee.wisc.edu`.

A database of aligned DNA sequences separated into groups is used as input to the program. First the function `TileSeqs` is used to pre-process the sequences into overlapping tiles, which will serve as the template DNA for probe design. Second, the `DesignProbes` function determines the set of all possible probes that meet certain design constraints, such as the ability to hybridize with the group of interest under specified

experimental conditions. Next, the complete set of probes is scored by its potential to cross-hybridize with sequences from other groups. Finally, the optimal set of dual probes is chosen that could be used in a FISH experiment to identify the microbes of interest.

2 The Objective of FISH Probe Design

The objective of FISH probe design is to balance sensitivity of the probe to the target group of sequences, while simultaneously maximizing specificity to prevent cross-hybridizations with other non-target groups. Here there is a fundamental trade-off, because if a probe is too weak it will not bind to its target, but if too strong it will also bind to non-targets. This balancing act is complicated by the difficulty of accurately predicting binding strength of the probe with a variety of potential target and non-target sequences. To confront this challenge, `DesignProbes` uses a state of the art model of probe hybridization in the presence of the denaturant formamide. This model, implemented in the function `CalculateEfficiencyFISH`, is used in conjunction with reasonable safety factors to minimize the need for experimental optimization.
In cases where the desired level of specificity cannot be achieved, `DesignProbes` predicts the degree of potential cross-hybridization with non-target groups. When a single probe is insufficient, dual probes with distinct colors can be employed to further increase specificity to the target group. `DesignProbes` will choose the optimal combination of dual probes that will minimize cross-hybridization overlap. During the experiment, the combination of the individual probe colors provides additional assurance of a positive identification. For example, if one probe is labeled green (e.g., fluorescein) and the other is labeled red (e.g., Cy3) then the hybridization of both probes at the same point would appear yellow.

3 Getting Started

3.1 Installing OligoArrayAux

The program OligoArrayAux (`http://mfold.rna.albany.edu/?q=DINAMelt/OligoArrayAux`) is used to predict hybridization efficiency and must be installed in a location accessible by the system. For example, the following code should print the installed OligoArrayAux version when executed from the *R* console:

```
> system("hybrid-min -V")


hybrid-min (OligoArrayAux) 3.8
By Nicholas R. Markham and Michael Zuker
Copyright (C) 2006
Rensselaer Polytechnic Institute
Troy, NY 12810-3590 USA
```

3.2 Startup

To get started we need to load the DECIPHER package, which automatically loads several other required packages.

```
> library(DECIPHER)
```

Help for the `DesignProbes` function can be accessed through:

```
> ? DesignProbes
```

If this vignette was installed on your system then the code in each example can be obtained by:

```
> browseVignettes("DECIPHER")
```

3.3 Creating a Sequence Database

We begin with a set of aligned 16S sequences representing a variety of bacteria collected from drinking water samples. We wish to design probes targeting only the genus *Sphingopyxis*, which is a common organism detected in drinking water distribution systems. This example uses a GenBank sequence file included as part of the DECIPHER package, but you could follow along with your own GenBank file of aligned sequences. Be sure to change the path names to those on your system by replacing all of the text inside quotes labeled "<<path to ...>>" with the actual path on your system.

```
> # specify the path to your sequence file:
> gb <- "<<path to GenBank file>>"
> # OR find the example sequence file used in this tutorial:
> gb <- system.file("extdata", "Bacteria_175seqs.gen", package="DECIPHER")
```

Next, there are two options for importing the sequences into a database: either save a database file or maintain the database in memory. Here we will build the database in memory because it is a small set of sequences and we do not intend to use the database later:

```
> # specify a path for where to write the sequence database
> dbConn <- "<<path to write sequence database>>"
> # OR create the sequence database in memory
> dbConn <- dbConnect(SQLite(), ":memory:")
> Seqs2DB(gb, "GenBank", dbConn, "Bacteria")
Reading GenBank file from line  1 to 1e+05

175 total sequences in the table DNA.
Time difference of 0.35 secs

[1] 175
```

3.4 Defining Groups

At this point we need to define groups of related sequences in the database we just created. In this case we wish to define phylogenetic groups based on the taxonomic information included in the original GenBank file. This rank information was stored by DECIPHER in the sequence database, and we can use it to identify the groups at the base taxonomic level (i.e., genus). Alternatively we could use the functions `FormGroups` or `IdClusters` to define groups.

```
> ids <- IdentifyByRank(dbConn, level=Inf, add2tbl=TRUE)
Updating column: "id"...
Formed 72 distinct groups.
Added to table DNA: "id".
Time difference of 0.03 secs
```

4 Probe Design Steps

4.1 Tiling Sequences

We continue by creating a set of k-mers, known as "tiles", that represent the sequences in each group. Here we must make several decisions that will affect probe design in the future. The defaults are to create tiles of

length 26-27 nucleotides with up to 10 permutations that represent at least 90% of the permutations present in each target site. These parameters will generally fit most probe designs, but it is recommended to read the help file for the `TileSeqs` function to make sure that it is doing what is desired. We will save the resulting tiles back to the database as a new table named "Tiles" in case we wish to access them in the future.

```
> tiles <- TileSeqs(dbConn, add2tbl="Tiles")


  |======================================================================| 100%

Time difference of 328.25 secs
```

4.2 Designing All Possible Probes

Next we wish to design probes targeting the 5 sequences identified as belonging to the genus *Sphingopyxis*. As an example, we will begin by designing probes for every possible target site that meet certain design criteria. By default, we allow up to 4 probe permutations as long as they cover 90% of the permutations present in their target site. Note that the input parameters to `DesignProbes` should be carefully considered in order adequately represent the experimental conditions that will be used. For example, here we limit the target sites to those between positions 120 and 1,450 because this is the region shared by most of the sequences.

```
> probes <- DesignProbes(tiles, identifier="Sphingopyxis",
         start=120, end=1450)


Sphingopyxis (678):
  |======================================================================| 100%

Time difference of 290 secs
```

We can now examine the top-scoring probe with maximal sensitivity and specificity to the target group.

```
> o <- order(probes$score, decreasing=TRUE)
> probes[o[1],]


      identifier start start_aligned permutations
826 Sphingopyxis   859          1019            3
           score                 probe.1
826 -1.43876.... CCGCGATTAGGATGTCAAACGCT
                    probe.2                 probe.3 probe.4
826 CCGCGATCAGGATGTCAAATGCT CCGCGATCAGGATGTCAAACGCT    <NA>
    efficiency.1 efficiency.2 efficiency.3 efficiency.4
826    0.5014489    0.5506237    0.6315190           NA
       FAm.1    FAm.2    FAm.3    FAm.4 coverage.1
826 35.04488 36.57339 39.17166       NA        0.4
    coverage.2 coverage.3 coverage.4
826        0.2        0.4         NA
mismatches
826 unclassified_Sphingomonadaceae (63.2%,0.00kcal/mol,
4.1%;CCGCGATCAGGATGTCAAACGCT/AGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (9.1%,6.20kcal/mol,
-17.8%;CCGCGATCAGGATGTCAAATGCT/AGCTTTTGACATCCCGGTCGCGG)
```

The best probe out of 678 candidate target sites has three permutations, with formamide melt points (FAm) between 35% and 39% [FA] (v/v). These three permutations can be represented with a single consensus probe:

```
> ConsensusSequence(DNAStringSet(probes[o[1], "probe"][1:3]))


Time difference of 0 secs

  A DNAStringSet instance of length 1
    width seq
[1]    23 CCGCGATYAGGATGTCAAAYGCT
```

Based on the column *mismatches*, this probe has two predicted cross-hybridizations with other groups in the sequence set. These groups, unclassified Sphingomonadaceae and unclassified Sphingomonadales, are predicted to have 63.2% and 9.1% hybridization efficiency at equilibrium in the hybridization buffer with 35% [FA] (v/v). The first non-target has exactly the same Gibb's free energy because it is a perfect match to the probe. Note that its formamide melt point (FAm) is 4.1% greater than the probe permutation with the lowest melt point, which is the limiting factor in the experiment. The second probe has 6.2 kcal/mol greater Gibb's free energy and a formamide melt point 17.8% [FA] less than the probe with the target. We can examine the predicted melt curve of the probes with the target and non-target:

```
> FA_range <- 0:70 # [FA] (%, v/v)
> probe <- probes$probe[o[1], 1:3]
> targets <- reverseComplement(DNAStringSet(probe))
> f <- function(FA)
        CalculateEfficiencyFISH(probe, targets,
                temp=46, P=250e-9, ions=1, FA)[, "HybEff"]
> efficiency <- matrix(unlist(lapply(FA_range, f)), ncol=3, byrow=TRUE)
> matplot(FA_range, efficiency, ylim=c(0,1), ylab="Hybridization Efficiency",
        xlab=expression(paste("[Formamide] (%, v/v)", sep="")),
        type="l", lwd=2, col="Blue", main="Formamide Curve", lty=1)
> nontargets <- DNAStringSet(c("AGCGTTTGACATCCTGATCGCGG",
        "AGCTTTTGACATCCCGGTCGCGG"))
> f <- function(FA)
        CalculateEfficiencyFISH(probe[3:2], nontargets,
                temp=46, P=250e-9, ions=1, FA)[, "HybEff"]
> efficiency <- matrix(unlist(lapply(FA_range, f)), ncol=2, byrow=TRUE)
> matlines(FA_range, efficiency, col="Red", lwd=2, lty=3)
> abline(h=0.5, lty=2, lwd=2, col="Orange")
> abline(v=35, lty=2, lwd=2, col="Green")
> legend("topright", legend=c("Targets", "Non-Targets", "50% Efficiency",
        "Experimental [FA]"), col=c("Blue", "Red", "Orange", "Green"),
        lwd=c(2, 2, 2, 2), lty=c(1, 3, 2, 2))
```

5 Dual Probe Design Steps

5.1 Designing the Optimal Dual Probe Set

Since no single probe was sufficient to completely distinguish all non-targets, we can further increase specificity by using two probes in combination. To do this we simply change the parameter *numProbeSets* to a

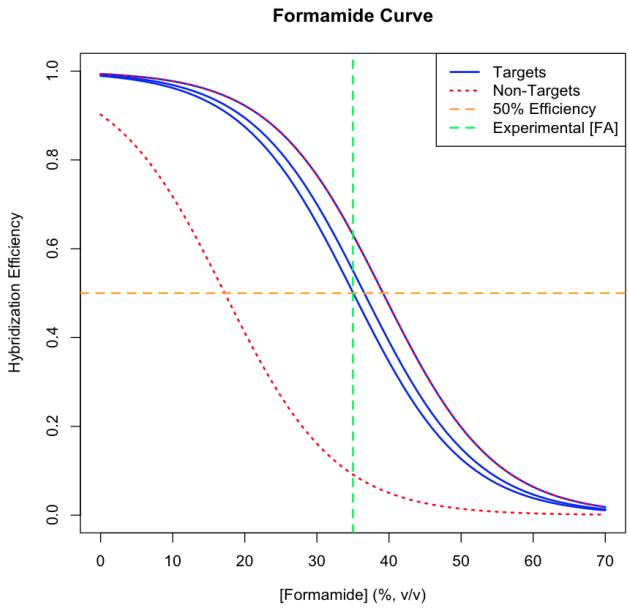

Figure 1: Formamide melt curves for the targets and non-targets

positive number from its default of zero. In doing so the `DesignProbes` function will search through potential combinations of two probes. The first and second probe's target sites must be separated by at least 50 nucleotides to mitigate helper-oligonucleotide effects. The two probes are then scored by their potential to cross-hybridize with the same non-targets to create a dual-color signal.

```
> probes <- DesignProbes(tiles, identifier="Sphingopyxis",
         start=120, end=1450, numProbeSets=100)


Sphingopyxis (678):
  |=================================================================================| 100%

Time difference of 300 secs
```

There are now multiple candidate dual probe sets for consideration, all of which are centered around the same variable regions in the alignment. Notice that the top dual probe set happens to include the best scoring single probe. Using two probes in conjunction was successful at eliminating all non-targets, as can be seen below in the empty column *mismatches_set*.

```
> head(probes)


    identifier start_one start_two start_aligned_one
1 Sphingopyxis       859       104              1019
2 Sphingopyxis       858       104              1018
3 Sphingopyxis       857       104              1017
4 Sphingopyxis       859       116              1019
5 Sphingopyxis       859       117              1019
6 Sphingopyxis       859       118              1019
```

```
  start_aligned_two permutations_one permutations_two
1               204               3               1
2               204               3               1
3               204               3               1
4               241               3               1
5               242               3               1
6               243               3               1
     score_one    score_two score_set
1 -1.43876.... -1.79020....         0
2 -1.44316.... -1.79020....         0
3 -1.44772.... -1.79020....         0
4 -1.43876.... -2.91132....         0
5 -1.43876.... -2.91132....         0
6 -1.43876.... -2.91132....         0
                one_probe.1               one_probe.2
1   CCGCGATTAGGATGTCAAACGCT   CCGCGATCAGGATGTCAAATGCT
2  CCGCGATTAGGATGTCAAACGCTG  CCGCGATCAGGATGTCAAATGCTG
3 CCGCGATTAGGATGTCAAACGCTGG CCGCGATCAGGATGTCAAATGCTGG
4   CCGCGATTAGGATGTCAAACGCT   CCGCGATCAGGATGTCAAATGCT
5   CCGCGATTAGGATGTCAAACGCT   CCGCGATCAGGATGTCAAATGCT
6   CCGCGATTAGGATGTCAAACGCT   CCGCGATCAGGATGTCAAATGCT
                one_probe.3 one_probe.4
1   CCGCGATCAGGATGTCAAACGCT        <NA>
2  CCGCGATCAGGATGTCAAACGCTG        <NA>
3 CCGCGATCAGGATGTCAAACGCTGG        <NA>
4   CCGCGATCAGGATGTCAAACGCT        <NA>
5   CCGCGATCAGGATGTCAAACGCT        <NA>
6   CCGCGATCAGGATGTCAAACGCT        <NA>
                two_probe.1 two_probe.2 two_probe.3
1 CATCCTTGGGCGATAAATCTTTGGT        <NA>        <NA>
2 CATCCTTGGGCGATAAATCTTTGGT        <NA>        <NA>
3 CATCCTTGGGCGATAAATCTTTGGT        <NA>        <NA>
4         GGCTCATCCTTGGGCGA        <NA>        <NA>
5         GGGCTCATCCTTGGGCG        <NA>        <NA>
6         CGGGCTCATCCTTGGGC        <NA>        <NA>
  two_probe.4 one_efficiency.1 one_efficiency.2
1        <NA>     0.501448....     0.550623....
2        <NA>     0.530359....     0.579076....
3        <NA>     0.675179....     0.716892....
4        <NA>     0.501448....     0.550623....
5        <NA>     0.501448....     0.550623....
6        <NA>     0.501448....     0.550623....
  one_efficiency.3 one_efficiency.4 two_efficiency.1
1     0.631518....               NA     0.580527....
2     0.658030....               NA     0.580527....
3     0.779824....               NA     0.580527....
4     0.631518....               NA     0.564271....
5     0.631518....               NA     0.622986....
6     0.631518....               NA     0.622986....
  two_efficiency.2 two_efficiency.3 two_efficiency.4
1               NA               NA               NA
```

```
2               NA               NA               NA
3               NA               NA               NA
4               NA               NA               NA
5               NA               NA               NA
6               NA               NA               NA
     one_FAm.1    one_FAm.2    one_FAm.3    one_FAm.4
1 35.04487.... 36.57338.... 39.17166....           NA
2 35.91037.... 37.38835.... 39.90073....           NA
3 40.30324.... 41.73392.... 44.16591....           NA
4 35.04487.... 36.57338.... 39.17166....           NA
5 35.04487.... 36.57338.... 39.17166....           NA
6 35.04487.... 36.57338.... 39.17166....           NA
     two_FAm.1    two_FAm.2    two_FAm.3    two_FAm.4
1 37.35508....           NA           NA           NA
2 37.35508....           NA           NA           NA
3 37.35508....           NA           NA           NA
4 37.51836....           NA           NA           NA
5 39.89269....           NA           NA           NA
6 39.89269....           NA           NA           NA
  one_coverage.1 one_coverage.2 one_coverage.3
1            0.4            0.2            0.4
2            0.4            0.2            0.4
3            0.4            0.2            0.4
4            0.4            0.2            0.4
5            0.4            0.2            0.4
6            0.4            0.2            0.4
  one_coverage.4 two_coverage.1 two_coverage.2
1             NA              1             NA
2             NA              1             NA
3             NA              1             NA
4             NA              1             NA
5             NA              1             NA
6             NA              1             NA
  two_coverage.3 two_coverage.4
1             NA             NA
2             NA             NA
3             NA             NA
4             NA             NA
5             NA             NA
6             NA             NA

1          unclassified_Sphingomonadaceae (63.2%,0.00kcal/mol,
4.1%;CCGCGATCAGGATGTCAAACGCT/AGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (9.1%,6.20kcal/mol,
-17.8%;CCGCGATCAGGATGTCAAATGCT/AGCTTTTGACATCCCGGTCGCGG)
2     unclassified_Sphingomonadaceae (65.8%,0.00kcal/mol,
4.0%;CCGCGATCAGGATGTCAAACGCTG/CAGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (10.2%,6.20kcal/mol,
-17.2%;CCGCGATCAGGATGTCAAATGCTG/CAGCTTTTGACATCCCGGTCGCGG)
3 unclassified_Sphingomonadaceae (78.0%,0.00kcal/mol,
3.9%;CCGCGATCAGGATGTCAAACGCTGG/CCAGCGTTTGACATCCTGATCGCGG)
```

```
unclassified_Sphingomonadales (17.2%,6.20kcal/mol,
-16.7%;CCGCGATCAGGATGTCAAATGCTGG/CCAGCTTTTGACATCCCGGTCGCGG)
4          unclassified_Sphingomonadaceae (63.2%,0.00kcal/mol,
4.1%;CCGCGATCAGGATGTCAAACGCT/AGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (9.1%,6.20kcal/mol,
-17.8%;CCGCGATCAGGATGTCAAATGCT/AGCTTTTGACATCCCGGTCGCGG)
5          unclassified_Sphingomonadaceae (63.2%,0.00kcal/mol,
4.1%;CCGCGATCAGGATGTCAAACGCT/AGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (9.1%,6.20kcal/mol,
-17.8%;CCGCGATCAGGATGTCAAATGCT/AGCTTTTGACATCCCGGTCGCGG)
6          unclassified_Sphingomonadaceae (63.2%,0.00kcal/mol,
4.1%;CCGCGATCAGGATGTCAAACGCT/AGCGTTTGACATCCTGATCGCGG)
unclassified_Sphingomonadales (9.1%,6.20kcal/mol,
-17.8%;CCGCGATCAGGATGTCAAATGCT/AGCTTTTGACATCCCGGTCGCGG)

1 Blastomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG) Novosphingobium (29.5%,2.96kcal/mol,
-8.7%;CATCCTTGGGCGATAAATCTTTGGT/TCCAAAGATTTATCGCCCAGGGATG)
Sphingomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG)
2 Blastomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG) Novosphingobium (29.5%,2.96kcal/mol,
-8.7%;CATCCTTGGGCGATAAATCTTTGGT/TCCAAAGATTTATCGCCCAGGGATG)
Sphingomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG)
3 Blastomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG) Novosphingobium (29.5%,2.96kcal/mol,
-8.7%;CATCCTTGGGCGATAAATCTTTGGT/TCCAAAGATTTATCGCCCAGGGATG)
Sphingomonas (44.7%,1.33kcal/mol,-3.9%;CATCCTTGGGCGATAAATCTTTGGT/
TCCAAAGATTTATCGCCCAAGGATG)
4 Blastomonas (56.4%,0.00kcal/mol,0.0%;GGCTCATCCTTGGGCGA/TCGCCCAAGGATGAGCC)
 Novosphingobium (40.2%,1.63kcal/mol,-6.4%;GGCTCATCCTTGGGCGA/TCGCCCAGGGATGAGCC)
Sphingomonas (56.4%,0.00kcal/mol,0.0%;GGCTCATCCTTGGGCGA/TCGCCCAAGGATGAGCC)
5 Blastomonas (62.3%,0.00kcal/mol,0.0%;GGGCTCATCCTTGGGCG/CGCCCAAGGATGAGCCC)
Novosphingobium (46.1%,1.63kcal/mol,-6.4%;GGGCTCATCCTTGGGCG/CGCCCAGGGATGAGCCC)
Sphingomonas (62.3%,0.00kcal/mol,0.0%;GGGCTCATCCTTGGGCG/CGCCCAAGGATGAGCCC)
6 Blastomonas (62.3%,0.00kcal/mol,0.0%;CGGGCTCATCCTTGGGC/GCCCAAGGATGAGCCCG)
Novosphingobium (46.1%,1.63kcal/mol,-6.4%;CGGGCTCATCCTTGGGC/GCCCAGGGATGAGCCCG)
Sphingomonas (62.3%,0.00kcal/mol,0.0%;CGGGCTCATCCTTGGGC/GCCCAAGGATGAGCCCG)
  mismatches_set
1
2
3
4
5
6
```

5.2 Visualizing the Target Sites

Often it is useful to visualize the target sites where the probes will hybridize on both the target and non-targets side-by-side. We can accomplish this by highlighting the region of both probes in the sequence set

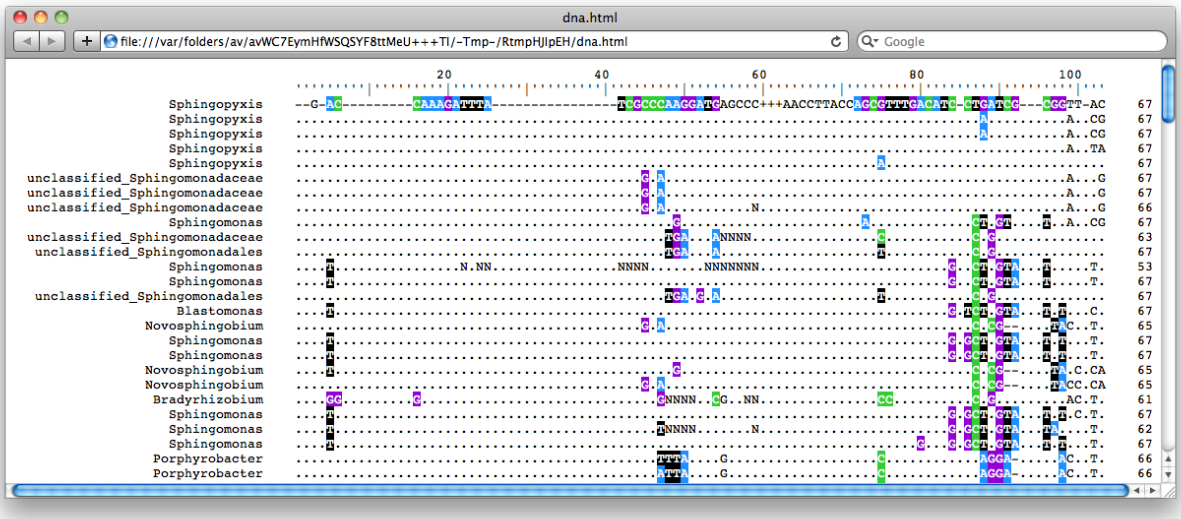

Figure 2: 16S sequences with target site for each probe colored

with `BrowseSequences`. After querying the database for the sequences of interest, we can selectively color the target site regions given by `start_aligned_one` and `start_aligned_two` outputs of `DesignProbes`. Next we name and order the sequences in the set such that the target (*Sphingopyxis*) appears at the top.

```
> dna <- SearchDB(dbConn, nameBy="id", verbose=FALSE)
> dbDisconnect(dbConn)
[1] TRUE
> # move the target group to the top of the sequence set
> w <- which(names(dna)=="Sphingopyxis")
> dna <- c(dna[w], dna[-w])
> BrowseSequences(dna, color=c(204, 253, 1019, 1045), highlight=1)
[1] TRUE
```

Examining the output, it is clear why these target sites were chosen for the dual probes: at least one of the dual probes has a number of mismatches to each of the non-target groups.

5.3 Finishing Up

Finally, we can order the first and second probe and try them out in an experiment! The probes should be labeled with different fluorphores (e.g., fluorescein and Cy3). In cases where all non-targets cannot be eliminated, it may be possible to further increase specificity by using a competitor oligonucleotide complementary to the non-target permutations given in the *mismatches* columns. Hybridization conditions could be further optimized based on the results of a formamide gradient experiment. If the two probes have markedly different optimal formamide concentrations then a dual-hybridization approach can be employed by washing with the more stringent condition followed by the second.

6 Session Information

All of the output in this vignette was produced under the following conditions:

- R version 3.0.2 (2013-09-25), `x86_64-unknown-linux-gnu`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, utils
- Other packages: BiocGenerics 0.8.0, Biostrings 2.30.0, DBI 0.2-7, DECIPHER 1.8.0, IRanges 1.20.0, RSQLite 0.11.4, XVector 0.2.0
- Loaded via a namespace (and not attached): KernSmooth 2.23-10, stats4 3.0.2, tools 3.0.2