Minimizing sets of enzymes to differentiate between species

David Buezas[‡]
CENTRIA [§]
FCT/UNL
david.buezas@gmail.com, João Almeida
CREM [¶]
FCT/UNL
jmfa@fct.unl.pt and Pedro Barahona
CENTRIA [§]
FCT/UNL
pb@di.fct.unl.pt

Abstract

A large number of species cannot be distinguished via standard non genetic analysis in the lab. In this paper we address the problem of finding minimum sets of restriction enzymes that can be used to unequivocally identify the species of a yeast specimen by analyzing the size of digested DNA fragments in gel electrophoresis experiments. The problem is first mapped into set covering and then solved using Constraint Programming techniques. Although the data sets used are relatively small (23 yeast species and 331 enzymes), a similar approach might be applicable to larger ones and to a number of variants as discussed in the conclusion. The subject of this paper has already raised the interest of our biologist partners and may become a benchmark for the application of Constraint Programming techniques to Bioinformatics.

1 Introduction

The problem of yeast identification was historically addressed through the study of both morphological traits and physiological features [3, 5, 16], but alternative molecular methods have been adopted to obtain the sequence of particular genomic regions and thus identify a given species [6, 11].

Although sequencing nucleic acids is more accessible than ever, it is still an expensive technique, especially if applied to a high numbers of specimens. In contrast to less expensive techniques like RFLP, RAPD, MSP-PCR (which allow the formation of clusters among the specimens to be identified, with inherent result limitations in scope), ARDRA [12] was proposed to differentiate between species of a eubacterial family and it represents an approach that goes beyond the mere clustering operation. The amplified fragment and the digestion products sizes are reproducible, characteristic for the substrate sequence, and thus characteristic for the source taxon, generally enabling the identification of the organism.

ARDRA-ITS [10] was developed with the purpose of differentiating fungal species. The differences between the original technique and the ITS variant lay on the primers, ITS1 and

[*] Centro de Inteligência Artificial, Dep. de Informática, Faculdade de Ciências e Tecnologia / Universidade Nova de Lisboa, Caparica, Portugal.

[†] Centro de Recursos Microbiológicos, Dep. Ciências da Vida, Faculdade de Ciências e Tecnologia / Universidade Nova de Lisboa, Caparica, Portugal.

[‡] The author David Buezas was supported by the European Master's Program in Computational Logic (EMCL).

[§] Centro de Inteligência Artificial, Dep. de Informática, Faculdade de Ciências e Tecnologia / Universidade Nova de Lisboa, Caparica, Portugal.

[¶] Centro de Recursos Microbiológicos, Dep. Ciências da Vida, Faculdade de Ciências e Tecnologia / Universidade Nova de Lisboa, Caparica, Portugal.

ITS4 [15], that amplify the 5.8S-ITS region of the operon, and in the set of enzymes used: HaeIII and TaqI. Other authors [1] took a step further and proposed the use of a variant, ARDRA-ITS, as an identification method for yeasts. They used a different set of restriction enzymes (CfoI, HaeIII, HinfI, and several more to resolve occasional ambiguities), and the latter target region, the 5.8S-ITS region. This genomic region also happens to be one of the better represented in the public nucleotidic sequences databases (GenBank, EMBL Bank and DDBJ consortium). This approach has been used with considerable success to identify yeasts associated with food [2, 7, 13] and a commercial database is available for this purpose (www.yeast-id.com), but its usefulness has been hindered by the reduced set of yeast strains studied and the limitations of size resolution of classical electrophoresis apparatus.

Recent papers are acknowledging the power of *in silico* contributions in this field. One is limited to the forecast of electrophoretic patterns [8], the other presents a program to assess the utility of a fixed set of endonucleases to distinguish between a given set of sequences [14]. However, the integration of all the available data in a comprehensive in silico approach, targeting optimality in identification by ARDRA is still to be proposed.

The purpose of this paper is twofold. Firstly, in the context of ARDRA-ITS, we propose to infer the minimum set of enzymes required to identify one, from a given set of yeasts. Secondly, we propose that this problem is used as benchmark for Constraint Programming methods applied to Bioinformatics. Although the instance presented in the paper has been solved, larger instances and variations of the problem may pose a relevant challenge to CP techniques.

The paper is organized as follows. Section 2 shows how the ARDRA-ITS technique can be cast into a minimum set covering problem. Section 3 presents different models to obtain both a single solution and all solutions to this problem, and briefly discusses the experimental results obtained with them. Finally, section 4 presents some initial conclusions and a discussion of further work.

2 Mapping ARDRA into a Minimum Set Covering problem

The ARDRA-ITS technique identifies one from a set of specimens through analysis of a specific DNA sub-sequence of its genome. Restriction enzymes (that, as is well known, cut double-stranded or single stranded DNA at specific recognition nucleotide sequences, known as *restriction sites*) play a central role in the ARDRA-ITS technique that proceeds as follows: First, a "standard" fragment of the test specimen DNA is obtained (in the case of yeasts, the 5.8S-ITS region of their operons), and many copies of it are produced. Secondly, a set of restriction enzymes are separately applied to these copies. The complete digestion of each enzyme yields several smaller nucleotide segments that, subject to gel-electrophoresis, originate bands of different lengths.

Each yeast - restriction enzyme pair generates a specific band pattern, but given the similarity of their DNA, several yeasts are likely to present similar patterns when digested by most restriction enzymes. Subject to some experimental error, there is a one-to-one correspondence between fragment sizes and the position of the respective band in the pattern, hence the sizes of the fragments obtained can be approximately calculated from the gel electrophoresis experiments. On the other hand, when its DNA sequence is known, the pattern produced by the digestion of yeast Y (or rather, the 5.8S-ITS region of its operon) by the restriction enzyme R can be computed by running a simulation of a gel electrophoresis experiment. A simple diagram of digestion in this context is shown in Figure 1.

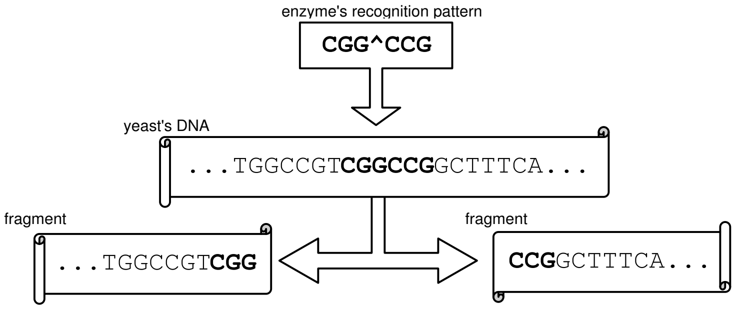

Figure 1: Diagram of digestion

A restriction enzyme R differentiates two yeast specimens Y_1 and Y_2 if the patterns it produces from them are distinguishable, i.e. at least one fragment in one of the digested yeasts is of a sufficiently different size from any fragment in the other digested yeast.

It is thus possible to produce a Boolean coverage table D, where rows denote yeast pairs and columns represent restriction enzymes. In this table the cell in the row $Y_i - Y_j$ and the column E_k tells whether that yeast pair is differentiated by that restriction enzyme. The problem of identifying a yeast among a set of similar yeasts can be formulated as finding a set S of restriction enzymes that differentiate any pair of yeasts, i.e. for all rows there is at least one enzyme in S such that its corresponding column has a true value for that row.

More formally, given a set of yeasts $Y = \{Y_1, ..., Y_{Ny}\}$ and a set of enzymes $E = \{E_1, ..., E_{Ne}\}$, we denote as $P(i,k)$ the induced pattern for Y_i by E_k, i.e. the set of segment lengths produced by the digestion of yeast Y_i by enzyme E_k. Two patterns P and Q are distinct if there is a fragment length in one of them that is sufficiently different (depending on the experimental error and denoted by $\not\approx$) from any fragment of the other pattern i.e.

$$\mathsf{distinct}(P,Q) =_{def} (\exists u \in P)(\forall v \in Q)(u \not\approx v) \vee (\exists u \in Q)(\forall v \in P)(u \not\approx v)$$

Two yeasts are differentiated by a restriction enzyme if the patterns induced in them are distinct:

$$(\forall i < j \text{ in } 1..N_y)(\forall k \text{ in } 1..N_e)$$
$$\mathsf{differentiate}(i,j,k) =_{def} \mathsf{distinct}(P(i,k), P(j,k))$$

A discriminating set of enzymes S is a subset of the set E of enzymes that, for any pair of yeasts in the set Y, has an element that differentiates them, i.e.

$$\mathsf{disc}(S,Y) =_{def} \forall (i\text{-}j) \in Y \;\; \exists k \in S : \mathsf{differentiate}(i,j,k)$$

A minimal (optimal) discriminating set of enzymes S is a discriminating set with minimal cardinality:

$$\mathsf{min\_disc}(S,Y) =_{def} \mathsf{disc}(S,Y) \wedge (\forall R \; \mathsf{disc}(R,Y) \rightarrow \#S \leq \#R)$$

Hence, given a set Y of yeast specimens, the ARDRA-ITS problem can be regarded as the problem of finding, from a set E of available restriction enzymes, a minimal discriminating set S for the set of yeast specimens.

Since some solutions might be preferred over others by the user (according to not yet fully formalized criteria such as reliability, availability and cost) it is also interesting to find not only one, but all minimal discriminating sets.

3 Alternative models

We tested a number of alternative models with a dataset of commercially available restriction enzymes, containing about 3500 elements [9]. As this set was redundant, meaning that many enzymes had the same recognition sequences, it was reduced to an equivalent one containing only 331 enzymes. The dataset of yeasts that we used is available at `http://www.cbs.knaw.nl/databases/`, it includes the nucleotide sequences of the 5.8S-ITS region of the operons of 23 yeast specimens. All the tests presented below where run in a Intel(R) Core(TM)2 Duo T5670 @1.80GHz (2 CPUs) with 3 GB of RAM, with a SICStus 4 CLP system.

3.1 Greedy model

A simple and greedy approach to solve the problem was implemented by accumulating the best enzymes (i.e. those that differentiate more yeast pairs still to be covered) until all yeast pairs are covered. The pseudo code is shown in Algorithm 1.

Algorithm 1 Greedy model

```
$S \leftarrow \emptyset$
$Y \leftarrow \{i\text{-}j : i \in 1..N_y,\ j \in 1..N_y,\ i < j\}$        ▷ the set of all yeast pairs to cover
$E \leftarrow \{k : k \in 1..N_e\}$        ▷ the set of all enzymes available
while $Y \neq \emptyset$ do
    $e \leftarrow \operatorname{argmax}_{k \in E} \left( \sum_{i\text{-}j \,\in\, Y} \text{differentiate}(i, j, k) \right)$        ▷ select the most covering enzyme
    $S \leftarrow S \cup \{e\}$
    $Y \leftarrow Y \backslash \{i\text{-}j : \text{differentiate}(i, j, e)\}$        ▷ subtract the covered yeast pairs
end while
```

Of course, this greedy approach does not guarantee that, upon termination, set S is an optimal discriminating set. In fact, notwithstanding the very fast execution time (125 ms), the solution found with our datasets contains nine enzymes, being far from minimal and therefore useless.

3.2 Backtrack model

This model guarantees optimality by finding differentiating sets of restriction enzymes with an increased size. The first set obtained is thus an optimal discriminating set. Alternative minimal differentiating sets can be obtained (with backtracking) by changing the condition in the while loop. The pseudo code is shown in Algorithm 2.

The execution time was close to 1 minute, but is heavily dependent on the order in which the enzymes are considered. If all solutions were to be found by backtracking alone, a huge number (around 6 million) of triplets would have to be tested, requiring an unacceptably huge execution time.

Algorithm 2 Backtrack model

```
$Y \leftarrow \{1, ..., N_y\}$                    ▷ the set of all yeast identifiers
$p \leftarrow 0$                                  ▷ $p$ stands for the size of the solution
$found \leftarrow$ **false**
**while** $\neg found$ **do**
    $p \leftarrow p + 1$                          ▷ the size of the optimal solution is searched incrementally
    **for all** $k_1..k_p \in 1..N_e : e_1 < ... < e_p$ **do**
        $S \leftarrow \{k_1, ..., k_p\}$
        **for all** $i, j \in 1..N_y : i < j$ **do**
            **if** $\neg$ discriminate$(i, j, S)$ **then**
                **break**
            **end if**
        **end for**
    **end for**
    $found \leftarrow$ differentiate$(Y, S)$
**end while**
```

3.3 Constraint Programming model with Boolean variables

The selection of the k^{th} enzyme in the discriminating set is modeled by a Boolean variable x_k, and a Constraint Programming system simply solves the problem of finding an assignment of these variables that covers all the yeast pairs, each covering being represented by a sum-product constraint of the variables x_k and the 0/1 constants representing the differentiating features. Of course all solutions can be obtained by backtracking. The pseudo code is shown in Algorithm 3.

Algorithm 3 Boolean CP model

```
$X \leftarrow [x_1, ..., x_{N_e}]$                ▷ one boolean variable for each enzyme identifier
**for all** $x \in 1.N_e$ **do**
    $x_k \in 0..1$
    **for all** $i, j \in 1..N_y : i < j$ **do**
        $\sum_{k \in 1..N_e} x_k *$ differentiate$(i, j, k) \geq 1$     ▷ constraints are imposed
    **end for**
**end for**
label$(X)$: minimising$\left(\sum_{k \in 1..N_e} x_k\right)$
```

With this model, the first minimum solution was found 15 seconds, which includes the 5 seconds necessary to initialize the covering table. This model is sufficiently efficient to compute all solutions of this problem instance. After the initialization time, all solutions were found in 15 minutes (the timing for finding the next solution vary widely from a some milliseconds to a few minutes).

3.4 Constraint Programming model with Finite Domain variables

Now each variable x_{ij} in the X vector is associated to the yeast pair Y_i-Y_j and its domain is the set of enzymes that differentiate such pair. By labeling X minimizing the number of different elements it uses, minimum solutions are found. The pseudo code is shown in Algorithm 4.

Algorithm 4 Finite Domain CP model I

```
$X = [x_{1\text{-}2}, ..., x_{i\text{-}j}, ..., x_{(N_y-1)\text{-}N_y}] : i < j$        ▷ one Finite Domain variable for each yeast pair
list_to_set($X, S$)
for all $i, j \in 1..N_y : i < j$ do
    $x_{i\text{-}j} \in \{k \in 1..Ne :$ differentiate$(i, j, k)\}$
                        ▷ the domain of $x_{i\text{-}j}$ is the set of enzyme identifiers that cover the $i$-$j$ pair
end for
label($X$): minimizing($\#S$)
```

To be effective, this model requires the minimization of the number of distinct values in list X (or equivalently, the number of elements in set S). In CP systems this can be achieved using the Nvalue(K,L) global constraint, that maps into the finite domain variable K, the number of distinct values in list L, as proposed in [4].

With this model, finding the first minimal solution takes 1 second (after the 5 seconds for table initialization). Unfortunately, the model cannot be used to find all solutions since many repetitions are obtained. For example, let us assume we have three yeast specimens (Y_1, Y_2, Y_3) forming three distinct yeast pairs

$$P_1 =< Y_1, Y_2 >, P_2 =< Y_1, Y_3 >, P_3 =< Y_2, Y_3 >$$

and that P_1 is covered by enzymes 2 and 3, P_2 by enzymes 1 and 3, and P_3 by enzymes 1 and 2. The tree yeast pairs would be represented as the vector $X = [x_{1\text{-}2}, x_{1\text{-}3}, x_{2\text{-}3}]$ where $x_{1\text{-}2} \in \{2, 3\}$, $x_{1\text{-}3} \in \{1, 3\}$ and $x_{2\text{-}3} \in \{1, 2\}$. This configuration allows six different labelings for X which use the least number of enzymes and therefore minimize the cardinality of S, namely:

$$\begin{array}{lll} X_1 = [2,1,1] & X_2 = [3,1,1] & X_3 = [3,3,2] \\ X_4 = [2,1,2] & X_5 = [3,3,1] & X_6 = [2,3,2] \end{array}$$

but since S is a set, each pair of labelings in the same column represent the same solution. Here there are only two repetitions per solution, but when real data is used the number of repetitions is so big that it prevents the enumeration of all solutions.

3.5 Avoiding repetitions with a different Finite Domain model

The previous model could not be easily adapted finding all solutions because, as just discussed, the same solution can come in a wide variety of encodings. Hence we decided to use a somewhat dual model of the previous one by having variables associated to restriction enzymes instead of yeast pairs. Once we found a minimal solution using the previous model, we may fix the size of a list of enzyme variables that must distinguish all yeast pairs. The pseudo code is shown in Algorithm 5.

Note that this model can only be setup when the size of a minimal solution is known. Alternatively, we may start with a set of enzymes with cardinality 1 and increment this size, as with the second (backtrack) model. With this model all solutions were found in 50 seconds. Hence, Finite Domain models improve on the Boolean model by one order of magnitude, both to find the first solution (1 sec against 10 secs) as well as all solutions (50 secs against 15 mins).

Algorithm 5 Finite Domain CP model II

```
$p \leftarrow \#S$                                   ▷ where #S is the minimum solution size
$S \leftarrow [k_1, ..., k_p]$
$k_1 < ... < k_p$                                    ▷ imposing an order avoids repeated solutions
**for all** $i, j \in 1..N_y : i < j$ **do**
    $E \leftarrow \{e :$ differentiate$(i, j, e)\}$
    $\bigvee_{i \in 1..p} (k_i \in S)$               ▷ a disjunctive constraint is posed
**end for**
label$(X)$
```

4 Conclusions and further work

In this paper we explore several potential models to a Bioinformatics problem, raised by the ARDRA-ITS experimental technique, requiring the minimization of the number of enzymes that must be used in gel electrophoresis experiments to unequivocally tell a yeast within a set of alternative and related yeasts specimens. By and large the problem can be applied to other types of organisms (ARDRA-ITS is an adaptation of ARDRA, originally used for identification of eubacterial family members) so its practical application can be quite wide.

Species are the taxonomic level we dealing with in this paper, but this approach can be extended to handle any taxonomic level. This idea is worth pursuing since when higher taxonomic levels are considered the execution time is reduced (because the number of specimen pairs in the coverage table is smaller) and solutions are likely to require a smaller number of enzymes.

The technique we used mapped the problem into a set covering problem, whose complexity is proportional to number of available restriction enzymes and the square of the number of specimen to identify. The data sets we used (around 300 enzymes and 23 yeasts, i.e. 253 yeast pairs) show the advantage of using constraint programming techniques over backtracking or purely heuristic search techniques, which solve this problem somewhat naively. Incidentally, this also justifies why we did not compare our models with Integer Programming alternatives, although we plan to do so whenever larger data sets are be available.

A number of variants to deal with uncertainty can be considered for this problem. On the one hand, we arbitrarily assumed that bands in electrophoresis experiments are distinguishable if their lengths differ by a certain minimum ratio (we used ± 5%). This is hardwired in our models but it would be interesting to model such relative difference as a parameter that is to be maximised, so that the solutions found are not only minimal but also the most reliable ones. On the other hand, we may consider that the yeast databases are obtained by consensus, and some of their nucleotides may vary. A quantified version of the problem would be to find a minimal set of enzymes that unequivocally identify a yeast, whatever the nucleotides a yeast variant may present. We plan to address both variants of this problem and provide a more comprehensive set of benchmarks, as well as experimental results.

References

[1] Esteve-Zarzoso B, Belloch C, Uruburu F, and Querol A, *Identification of yeasts by RFLP analysis of the 5.8s rRNA gene and the two ribosomal internal transcribed spacers*, International Journal of Systematic Bacteriology **49** (1999), 329–337.

[2] Esteve-Zarzoso B, Fernandez-Espinar MT, and Querol A, *Authentication and identification of saccharomyces cerevisiae 'flor' yeast races involved in sherry ageing*, Antonie Van Leeuwenhoek International Journal of General and Molecular Microbiology **85** (2004), 151–158.

[3] J. Barnett, R. Payne, and D. Yarrow, *Yeasts, characteristics and identification*, 3rd ed., Cambridge University Press, 2000.

[4] Christian Bessiere, Emmanuel Hebrard, Brahim Hnich, Zeynep Kiziltan, and Toby Walsh, *Filtering algorithms for the NValue constraint*, Constraints **11** (2006), no. 4, 271–293.

[5] K. Boundy-Mills, *The yeast handbook, biodiversity and ecophysiology of yeasts* (2006), 67–100.

[6] C.P. Kurtzman and C.J. Robnett, *Identification and phylogeny of ascomycetous yeasts from analysis of nuclear large subunit (26s) ribosomal DNA partial sequences*, Antonie van Leeuwenhoek **73** (1998), 331–371.

[7] Elena S. Naumova, Nataliya N. Sukhotina, and Gennadi I. Naumov, *Molecular-genetic differentiation of the dairy yeast kluyveromyces lactis and its closest wild relatives*, FEMS Yeast Research **5** (2004), no. 3, 263–269.

[8] Raspor P, Zupan J, and Cadez N, *Validation of yeast identification by in silico RFLP*, Journal of Rapid Methods and Automation in Microbiology **15** (2007), 267–281.

[9] R.J. Roberts, T. Vincze, J. Posfai, and D. Macelis, *REBASE–a database for DNA restriction and modification: enzymes, genes and genomes*, Nucleic Acids Res **38** (2010), 234–236.

[10] O. Schmidt and U. Moreth, *Genetic studies on house rot fungi and a rapid diagnosis*, European Journal of Wood and Wood Products **56** (1998), no. 6, 421–425.

[11] G. Scorzetti, J.W. Fell, A. Fonseca, and A. Statzell-Tallman, *Systematics of basidiomycetous yeasts: a comparison of large subunit d1/d2 and internal transcribed spacer rDNA regions*, FEMS yeast research **2** (2002), 495–517.

[12] M. Vaneechoutte, R. Rossau, P. De Vos, M. Gillis, D. Janssens, et al., *Rapid identification of bacteria of the Comamonadaceae with amplified ribosomal DNA restriction analysis (ARDRA)*, FEMS Microbiol Lett **72** (1992), 227–233.

[13] Bockelmann W, Heller M, and Heller K, *Identification of yeasts of dairy origin by amplified ribosomal DNA restriction analysis (ARDRA)*, Int Dairy J **18** (2008), 1066–1071.

[14] Wei W, Lee IM, Davis RE, Suo X, and Zhao Y, *Automated RFLP pattern comparison and similarity coefficient calculation for rapid delineation of new and distinct phytoplasma 16sr subgroup lineages*, International Journal of Systematic and Evolutionary Microbiology **58** (2008), 2368–2377.

[15] T. White, T. Bruns, S. Lee, and J. Taylor, *Amplification and direct sequencing of fungal ribosomal RNA genes for phylogenetics* (1990), 315–322.

[16] D. Yarrow, *The yeasts, a taxonomic study. methods for the isolation, maintenance and identification of yeasts* (1998), 148–152.