BIOINFORMATICS

Vol. 17 no. 3 2001
Pages 214–225

Efficient primer design algorithms

Thomas Kämpke[1], Markus Kieninger[2] and Michael Mecklenburg[3]

[1]Forschungsinstitut für anwendungsorientierte Wissensverarbeitung FAW, Helmholtzstr. 16, 89081 Ulm, Germany, [2]jSoft GmbH, Bahnhofstr. 7, 73447 Oberkochen, Germany and [3]INTERACTIVA Bioteknik AB & BT Biomedical Technology, IDEON Research Park, Ole Römersväg 12, S 223 70 Lund, Sweden

Received on March 8, 2000; revised on September 14, 2000; accepted on October 19, 2000

ABSTRACT

Motivation: Primer design involves various parameters such as string-based alignment scores, melting temperature, primer length and GC content. This entails a design approach from multicriteria decision making. Values of some of the criteria are easy to compute while others require intense calculations.

Results: The reference point method was found to be tractable for trading-off between deviations from ideal values of all the criteria. Some criteria computations are based on dynamic programs with value iteration whose run time can be bounded by a low-degree polynomial. For designing standard PCR primers, the scheme offers in a relative gain in computing speed of up to 50 : 1 over ad-hoc computational methods. Single PCR primer pairs have been used as model systems in order to simplify the quantization of the computational acceleration factors. The program has been structured so as to facilitate the analysis of large numbers of primer pairs with minor modifications. The scheme significantly increases primer design throughput which in turn facilitates the use of oligonucleotides in a wide range of applications including: multiplex PCR and other nucleic acid-based amplification systems, as well as in zip code targeting, oligonucleotide microarrays and nucleic acid-based nanoengineering.

Availability: A public version of the software DOPRIMER is accessible under http://doprimer.interactiva.de

Contact: kaempke@faw.uni-ulm.de

INTRODUCTION

The development of polymerase chain reaction (PCR) has revolutionized genetic analysis and engineering science. Other nucleic acid-based amplification techniques have been subsequently developed including ligase chain reaction, strand displacement amplification, and the RNA-based 3SR amplification methods. All these techniques make use of the ability of short synthetic nucleotides, also called oligonucleotides to specifically hybridize to complementary sequences.

The objective of PCR is to amplify a specific DNA fragment, the target sequence. Primers function in pairs, the so-called forward primer and the so-called reverse primer with this distinction being arbitrary from the computational viewpoint. The primer pairs are chosen such that they will be extended towards each other to cover the given target region, see Figure 1 (left). PCR begins with a high temperature (95 °C) denaturation step converting the double-stranded DNA into single-stranded DNA, followed by a low temperature step (45–65 °C) during which the primers hybridize and finally an intermediate temperature step (72 °C) for the primer extension. Typically 25–45 of these cycles are performed.

Formally, primers are considered as strings over the alphabet $\Sigma = \{A, C, G, T\}$ with the set of all these strings being Σ^*. As usual, the first position of a primer is denoted by 5′ end while the terminating position is denoted 3′ end. Each primer will be chosen within a window whose length and location is subject to specification; windows are not shown in Figure 1 but will be illustrated later. The final situation of PCR is shown in Figure 1 (right).

Primer assessment extends beyond string matching and involves criteria including the proximity between primer melting temperatures, minimization of hybridization effects between forward and reverse primers, and avoidance of hybridization of primers with themselves. The latter two criteria are dealt with by annealing values. The design complexity increases in so-called multiplex PCR. This involves performing multiple PCR reactions simultaneously in a single tube. Consequently, this requires that physical parameters such as cycle number, cycle duration and annealing temperature are identical for all of the PCR reactions. Moreover, the analysis of unintended primer–primer interactions becomes more intricate.

All primer criteria are real-valued. Typically, no primer pair is consistently better than all others entailing trade-offs between the criteria. The ideal point method (Yu, 1989) which is applied here to 12 criteria for PCR requires establishing ideal values for all the criteria. Actual

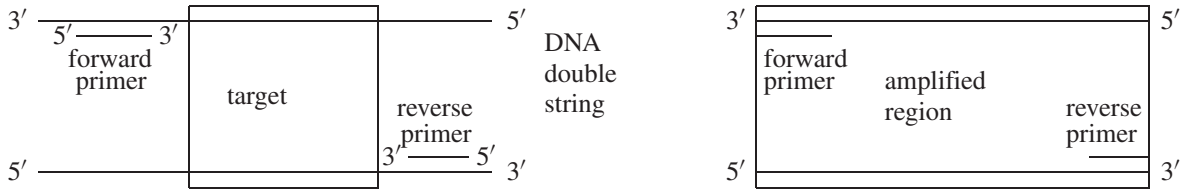

Fig. 1. Target enclosed by pair of primers (left) and ideal terminating situation (right).

differences are then aggregated by weighted sums. The design of primers is facilitated by constructing candidates and selecting the best. To reduce the space and hence the effort of that search, some criteria may be externally constrained.

Another major aim of these studies was to improve throughput by increasing the speed at which primer annealing computations could be performed. Dynamic programming will keep the overall computation effort at a tractable level. The use of dynamic programs for biological sequence alignment dates back to Waterman (1984). In contrast to ordinary alignment problems, our scheme addresses alignment as well as primer design issues.

The paper is organized as follows. The assessment of a primer pair is presented in the next section. Both an exact solution as well as faster approximating solutions are presented in the following section. Even faster approximations result from externally specified bounds on the feasibility of primer candidates. Assessment of a set of primer pairs for multiplex PCR is presented in the fourth section and respective search algorithms are given in the fifth section. The penultimate section discusses implementation issues both from an algorithmic as well as an empirical perspective. And the final section provides some conclusions and future applications.

$|A|$ denotes the number of elements of a finite set A, and $|p|$ denotes the length of a string p. $O(f(n))$ denotes functions g such that $g(n) \leqslant C \cdot f(n)$ for all n and $\Theta(f(n))$ denotes functions g such that $c_1 \cdot f(n) \leqslant g(n) \leqslant c_2 \cdot f(n)$ for all n.

ASSESSMENT OF INDIVIDUAL PRIMERS AND PRIMER PAIRS

Primer design is based on the subsequent criteria including various annealing values. The annealing values neither adhere to the frequently propagated edit distance or Levenshtein metric nor to a generalization thereof (cf. Apostolico and Galil, 1997). The reason for rejecting the edit distance in conjunction with primers is that insertions and deletions will lead to primer infeasibility for PCR rather than to situations which can be tolerated at some formal cost.

Melting temperature and GC content

A proper computation of the primer melting temperature does not appear to exist. A prominent approximation for the melting point of primer $p = (p_1, \ldots, p_n)$ (cf. Rychlik and Rhoads, 1989; Borer *et al.*, 1974; Breslauer *et al.*, 1986; Freier *et al.*, 1986) is the formula

$$T_{m,1}(p) = \frac{\triangle H(p)}{\triangle S(p) + R \cdot \ln(\frac{\gamma}{4})} + T_0 + t,$$

where $R = 1.987$ (cal/°C∗mol) is the molar gas constant, $\gamma = 50 \cdot 10^{-9}$ is the dimensionless (molar) concentration of the primer in its solution, $T_0 = -273.15\,°C$, and $t = -21.6\,°C$ is an empirical temperature correction. The value for t may depend upon the ion concentration and other unknown factors. $\triangle H(p)$ and $\triangle S(p)$ are the enthalpy and entropy of p which are computed according to the nearest neighbour schemes $\triangle H(p) = \sum_{i=1}^{n-1} \triangle H(p_i, p_{i+1})$ and $\triangle S(p) = \sum_{i=1}^{n-1} \triangle S(p_i, p_{i+1})$, where entropy and enthalpy of a string consisting of two bases—a duplex—is given as follows (Breslauer *et al.*, 1986):

Nearest neighbour thermodynamics

(p_i, p_{i+1})	$\triangle H(p_i, p_{i+1})$	$\triangle S(p_i, p_{i+1})$
AA or TT	9.1	24.0
AT	8.6	23.9
TA	6.0	16.9
CA or TG	5.8	12.9
GT or AC	6.5	17.3
CT or AG	7.8	20.8
GA or TC	5.6	13.5
CG	11.9	27.8
GC	11.1	26.7
GG or CC	11.0	26.6

All values refer to the energy required to disrupt the hydrogen bonds of a single base pair of a paired chain. This is assumed to be influenced by neighbouring bases. Values refer to the concentration of 1 M = 1 mol NaCl per l at 25 °C, and pH 7. The unit of $\triangle H$ is kcal/mol, whereas the unit of $\triangle S$ is cal/K per mol (1 cal = 4.184 J), see Breslauer *et al.* (1986, p. 3748). An example of

nearest neighbour thermodynamics is $p = GGAT$ with $\triangle H(GGAT) = \triangle H(GG) + \triangle H(GA) + \triangle H(AT) = 11.0+5.6+8.6 = 25.2$ (kcal/mol). There are several other tables of nearest neighbour thermodynamics available in the literature. For a detailed discussion see Owczarzy *et al.* (1998).

A simple approximation of the melting temperature in °C proposed here is $T_{m,2}(p) = 4 \cdot \#G\ in\ p + 4 \cdot \#C\ in\ p + 2 \cdot \#A\ in\ p + 2 \cdot \#T\ in\ p$. This formula was empirically derived by determining the melting temperatures of numerous primers and is valid for primers whose length lies in the interval 16–28 nucleotides in length.

The important criterion GC content simply measures the percentage of G and C of the primer

$$GC(p) = \frac{\#G\ in\ p + \#C\ in\ p}{|p|} \cdot 100.$$

The motivation for considering this quantity is the presence of three hydrogen bonds in GC pairs as compared with only two for AT pairs.

Self annealing and self-end annealing

Each primer is tested for unintended hybridization with itself by testing for self annealing and for self-end annealing (Hillier and Green, 1991). There is no 'self-begin annealing' test since extension always occurs at the 3′ end of a nucleotide sequence. The test for self annealing also accounts for the primer–dimer effect which is hybridization of one part of a primer molecule with another part.

The tests are based on a real-valued function S depending on string alignments with overlapping regions. Let $x = (x_1, \ldots, x_n) \in \Sigma^*$ and $y = (y_1, \ldots, y_m) \in \Sigma^*$. For the sake of simplicity of formulas, one of the strings, say y, is assumed to be enlarged at both ends by sufficiently many improper characters such as the 'empty' symbol $E \notin \Sigma$. The score of a pair of characters is defined to be

$$s(x_i, y_j) = \begin{cases} 2, & \text{if} \quad \{x_i, y_j\} = \{A, T\} \\ 4, & \text{if} \quad \{x_i, y_j\} = \{C, G\} \\ 0, & \text{else.} \end{cases}$$

Here, $\{A, T\}$ and $\{C, G\}$ denote two-element sets. The character scoring function s is extended to the alignment value

$$S(x, y) = \max_{k=-(n-1),\ldots,m-1} \sum_{i=1}^{n} s(x_i, y_{i+k}).$$

The intuition is that string x is translated by k positions relative to the 'initial' alignment with x_1 corresponding to y_1, x_2 corresponding to y_2, etc. Each alignment with at least one overlapping position of x and the proper part of y is considered. This corresponds to the translations $k = -n + 1, \ldots, m - 1$. Function S is symmetric, i.e. $S(x, y) = S(y, x)$.

Function S is now modifed to admit only alignments such that x_1 or y_m belongs to the overlapping region. Also, scoring counts from these characters onwards in an uninterrupted fashion where the latter means that a term contributing zero to the sum will stop summation. Formally, this score is denoted by $S''(x, y)$. The number of sums over which the maximum for the restricted alignment value S'' is actually taken depends on the sequences x and y. Function S'' is generally not symmetric. The two restrictions from S to S'' immediately imply $S''(x, y) \leqslant S(x, y)$.

Self annealing of a primer $p = (p_1, \ldots, p_n)$ is tested by aligning p with itself in the opposite direction since nucleotide chains bond this way if they do at all; the 5′ end of one string is aligned with the 3′ end of the other, cf. Figure 1. With the reverse string $\tilde{p} = (p_n, \ldots, p_1)$ this leads to the self annealing score

$$\mathrm{sa}(p) = S(\tilde{p}, p).$$

The reverse of a primer is not to be confused with the reverse primer of a primer pair. Evidently, a primer of length n has the $2n - 1$ overlapping alignments $k = -(n - 1), \ldots, n - 1$.

The test for self end annealing considers only those alignments for which the 3′ end belongs to the overlapping region. Only subsequences of uninterrupted bonds beginning at the 3′ ends are considered for self-end annealing. Formally, the self end annealing value of primer p is defined to be

$$\mathrm{sea}(p) = S''(\tilde{p}, p).$$

EXAMPLE 1. Let $p = GATTA$. Length $n = 5$ results in 9 overlapping alignments arranged in the order of $k = -4, \ldots, 4$. Vertical bars denote a positive score of a character pair.

```
    5'-GATTA-3'          5'-GATTA-3'          5'-GATTA-3'
           -> 0                 -> 0                 -> 0
3'-ATTAG-5'          3'-ATTAG-5'          3'-ATTAG-5'

  5'-GATTA-3'          5'-GATTA-3'          5'-GATTA-3'
    ||         -> 4      | |         -> 4                -> 0
3'-ATTAG-5'          3'-ATTAG-5'          3'-ATTAG-5'

5'-GATTA-3'          5'-GATTA-3'          5'-GATTA-3'
  | |          -> 4      ||          -> 4                -> 0
3'-ATTAG-5'          3'-ATTAG-5'          3'-ATTAG-5'
```

The empty symbol E is omitted throughout. The self annealing value is $\mathrm{sa}(GATTA) = \max\{0, 0, 0, 4, 4, 0, 4, 4, 0\} = 4$. The contribution of each alignment to the maximum is specified above together with the alignments.

From the five alignments $k = 0, \ldots, 4$ with the $3'$ end lying in the overlap only the two alignments for $k = 3$ and $k = 4$—on the left and center of the bottom row—do not begin with a zero count. Thus, only these two contribute to the self-end annealing score. The alignment $k = 3$ scores 2 and the alignment $k = 4$ scores $2 + 2$ resulting in $\mathrm{sea}(GATTA) = \max\{2, 4\} = 4$.

Criteria for primer pairs

Primer pairs are tested for unintended hybridization with each other. This test again consists of two modes that are called pair annealing and pair-end annealing. The modes are similar to the self and self-end annealing tests. Both primers $p = (p_1, \ldots, p_n)$ and $q = (q_1, \ldots, q_m)$ are arranged in all overlapping antithetic alignments and maximum scores are taken. Formally,

$$\mathrm{pa}(p, q) = S(\tilde{p}, q).$$

Pairwise annealing values generalize the self annealing values in the sense of $\mathrm{pa}(p, p) = \mathrm{sa}(p)$. The test for end annealing involves only alignments with at least one of the $3'$ end belonging to the overlapping region and only subsequences of uninterrupted bonds beginning at one of the $3'$ ends are considered for the pair end annealing value of p and q,

$$\mathrm{pea}(p, q) = S''(\tilde{p}, q).$$

Though S'' is not symmetric, function pea is, i.e. $\mathrm{pea}(p, q) = \mathrm{pea}(q, p)$.

EXAMPLE 2. A typical primer pair (p, q) consisting of a forward primer p and a reverse primer q is given with individual assessments as follows.

Forward primer	
p	GGATTGATAATGTAATAGG
$\lvert p\rvert$	19
$GC(p)$ [in %]	32
$T_{m,1}(p)$ [in °C]	38
$\mathrm{sa}(p)$	12
$\mathrm{sea}(p)$	0
Reverse primer	
q	CATTATGGGTGGTATGTTGG
$\lvert q\rvert$	20
$GC(q)$ [in %]	45
$T_{m,1}(q)$ [in °C]	50
$\mathrm{sa}(q)$	20
$\mathrm{sea}(q)$	4

The alternative temperature values are $T_{m,2}(p) = 50\,°\mathrm{C}$ and $T_{m,2}(q) = 58\,°\mathrm{C}$. The common assessment of the primer pair is as follows

(p, q)	(GGATTGATAATGTAATAGG,CATTATGGGTGGTATGTTGG)
$\mathrm{pa}(p, q)$	16
$\mathrm{pea}(p, q)$	4

Two sample computations are $GC(p) = \frac{6}{19} \cdot 100 \approx 32$ and

$$T_{m,1}(p) = \frac{(11.0 + \cdots + 7.8 + 11.0)\mathrm{kcal/mol}}{(26.6 + \cdots + 20.8 + 26.6)\mathrm{cal/(°C \cdot mol)} + 1987\mathrm{cal/(°C \cdot mol)} \cdot \ln\left(\frac{50 * 10^{-9}}{4}\right)} - 273,15\,°\mathrm{C} - 21,6\,°\mathrm{C}$$
$$= \frac{138.8 \cdot k\mathrm{cal/mol}}{362.4\mathrm{cal/(°C \cdot mol)} + 1987\mathrm{cal/(°C \cdot mol)} \cdot \ln\left(\frac{50 * 10^{-9}}{4}\right)} - 273,15\,°\mathrm{C} - 21,6\,°\mathrm{C} = 38\,°\mathrm{C}.$$

The self annealing value of p is $\mathrm{sa}(p) = 12$ which is attained by several alignments including the following.

```
5'-GGATTGATAATGTAATAGG-3'
       | | | | | |         -> 12
3'-GGATAATGTAATAGTTAGG-5'
```

The self-end annealing value of p is $\mathrm{sea}(p) = 0$. This is obvious, because p has a G at both ends and there is no C in p. The pair annealing value of p and q is $\mathrm{pa}(p, q) = 16$ and the pair-end annealing value is $\mathrm{pea}(p, q) = 4$ which are attained by the subsequent alignments respectively.

```
      5'-GGATTGATAATGTAATAGG-3'
            |   ||||||          -> 16
3'-GGTTGTATGGTGGGTATTAC-5'

 5'-GGATTGATAATGTAATAGG-3'
                   |           -> 4
3'-GGTTGTATGGTGGGTATTAC-5'
```

In the sequel any of the formulas for melting temperatures will be used with values being denoted $T_m(p)$, $T_m(q)$, etc. Other criteria can be taken into consideration like testing for unintended hybridization within the target area such that the alignment score of the primer and a target string is as low as possible. Another criterion is the number of 'GC' substrings in one primer indicating stable bonding. A third criterion could consist of testing the primer for ending with 'GC' which may be considered to be favourable.

Aggregation of assessments for ordinary PCR

A primer pair (p, q) is assigned the scoring vector

$$\mathrm{sc}(p, q) = (|p|, |q|, GC(p), GC(q), T_m(p), T_m(q), \mathrm{sa}(p), \mathrm{sa}(q), \mathrm{sea}(p), \mathrm{sea}(q), \mathrm{pa}(p, q), \mathrm{pea}(p, q))^{\mathrm{T}} \in \mathbb{R}^{12}.$$

All primers are designed to have ideal values of length, GC content, and melting temperature which are specified

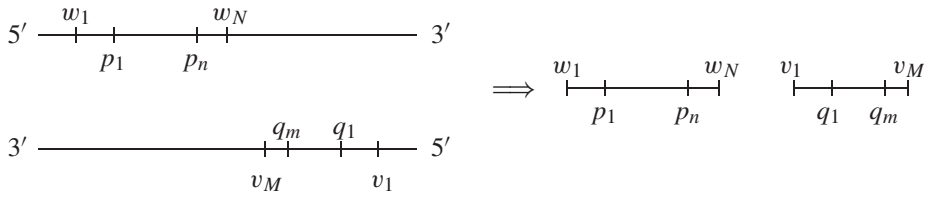

Fig. 2. Ignoring the double string character and the antithetic alignment within DNA.

externally by the designer of the hybridization experiment. These ideal values are to be specified for forward and reverse primers. The ideal score vector or reference vector for the primer pair is

$$\mathrm{sc}_{\mathrm{ideal}} = (\mathrm{length}_f, \mathrm{length}_r, GC_f, GC_r, T_{m,f}, T_{m,r}, 0, 0, 0, 0, 0, 0)^{\mathrm{T}}.$$

All ideal annealing values are set to zero and typically $T_{m,f} = T_{m,r}$ as well as $GC_f = GC_r$. The final assessment of a primer pair (p, q) can be its deviation from the reference in terms of the l_1-distance $||\,\mathrm{sc}(p,q) - \mathrm{sc}_{\mathrm{ideal}}\,|| = \sum_{i=1}^{12} |\,\mathrm{sc}(p,q)_i - (\mathrm{sc}_{\mathrm{ideal}})_i\,|$. Here, we employ a weighted distance $||\,\mathrm{sc}(p,q) - \mathrm{sc}_{\mathrm{ideal}}\,||_\kappa = \sum_{i=1}^{12} \kappa_i\,|\,\mathrm{sc}(p,q)_i - (\mathrm{sc}_{\mathrm{ideal}})_i\,|$ with weights given in the following table.

Deviation from ideal length, $\mathrm{length}_f = \mathrm{length}_r$	$\kappa_1 = \kappa_2 = 0.5$
Deviation from ideal GC content, $GC_f = GC_r$	$\kappa_3 = \kappa_4 = 1$
Deviation from ideal temperature, $T_{m,f} = T_{m,r}$	$\kappa_5 = \kappa_6 = 1$
Deviation from ideal self annealing value 0	$\kappa_7 = \kappa_8 = 0.1$
Deviation from ideal self-end annealing value 0	$\kappa_9 = \kappa_{10} = 0.2$
Deviation from ideal pair annealing value 0	$\kappa_{11} = 0.1$
Deviation from ideal pair-end annealing value 0	$\kappa_{12} = 0.2$

Computing the l_1-distance or a weighted version thereof requires $O(n^2 + m^2 + nm) = O((n+m)^2)$ time which essentially is the $O(n^2)$ complexity for computing $\mathrm{sa}(p)$ and $\mathrm{sea}(p)$, the $O(m^2)$ complexity for computing $\mathrm{sa}(q)$ and $\mathrm{sea}(q)$, and the $O(nm)$ complexity for computing $\mathrm{pa}(p,q)$ and $\mathrm{pea}(p,q)$.

The space $\mathbb{R}^{12}$ is not endowed with monotonicity with respect to the assessment criteria. Though this is true for the last six coordinates where a smaller alignment score is preferable to a larger one with all other criteria being equal, there is no monotonicity in primer length, GC content, and melting temperature. Thus, the ideal point cannot be preplaced without another assumption.

PCR PRIMER COMPUTATION

Given two windows $w = (w_1, \ldots, w_N)$ and $v = (v_1, \ldots, v_M)$ the objective is to find substrings $p = (p_1, \ldots, p_n) = (w_i, \ldots, w_{i+n-1})$ and $q = (q_1, \ldots, q_m) = (v_j, \ldots, v_{j+m-1})$ which serve as primers. Windows are specified externally (see Introduction). Substrings of windows are required to have at least length two in order to avoid trivial complications. Though the windows lie on different strings of the DNA, the antithetic alignment of both strings can be ignored for computational purposes, see Figure 2. The sets of all such substrings of w and v are denoted $\mathrm{Sub}(w)$ and $\mathrm{Sub}(v)$ respectively. The PCR primer computation is formalized as the reference point approximation problem

$$\min_{p \in \mathrm{Sub}(w),\, q \in \mathrm{Sub}(v)} ||\,\mathrm{sc}(p,q) - \mathrm{sc}_{\mathrm{ideal}}\,||_\kappa .$$

This optimization problem is a multicriteria decision problem with additive value function $-\,||\,\mathrm{sc}(p,q) - \mathrm{sc}_{\mathrm{ideal}}\,||_\kappa$ where a so-called alternative (p,q) with maximum value is to be chosen, (see Keeney and Raiffa, 1976). The primer computation problem can be solved in polynomial time since $|\,\mathrm{Sub}(w)\,| = N(N-1)/2$ and $|\,\mathrm{Sub}(v)\,| = M(M-1)/2$. Thus, complete enumeration leads to $O(N^2M^2)$ candidate pairs each requiring $O((n+m)^2) = O((N+M)^2)$ computational effort. The latter results from the computation effort for $\mathrm{sa}(p)$, $\mathrm{sea}(p)$, $\mathrm{sa}(q)$, $\mathrm{sea}(q)$, $\mathrm{pa}(p,q)$, and $\mathrm{pea}(p,q)$ which is $2O(n^2) + 2O(m^2) + 2O(nm) = O((n+m)^2)$. This results in an $O((N^2M + M^2N)^2)$ overall computation bound. More efficient computations are based on dynamic programming in the form of value iteration.

If a primer pair is rejected due to reasons beyond the formal criteria, the next best primer pair can be selected. Thus, either all or some of the highest ranking primer pairs are sorted in order of weighted distance from the ideal vector. This will especially matter in the multiplex case.

Dynamic programming requisites

As a preparatory step, $S(x^+, y)$ is assumed to be computed for $x^+ = (x_1, \ldots, x_{n+1})$, where $S(x,y)$ and suitably selected intermediate values are supposed to be known. The latter comprise $S_k(x,y) = \sum_{i=1}^{n} s(x_i, y_{i+k})$ for $k = -n+1, \ldots, m-1$. Then

$$S_k(x^+, y) = \begin{cases} S_k(x,y) + s(x_{n+1}, y_{n+1-k}) & \text{for } k = -n+1, \ldots, m-1 \\ S_{-n}(x^+, y) = s(x_{n+1}, y_1) & \text{for } k = -(n+1)+1 = -n. \end{cases}$$

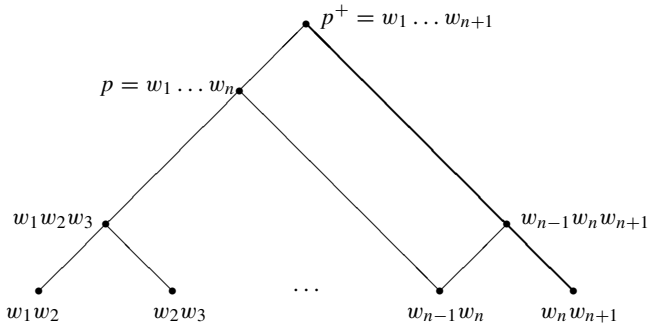

Fig. 3. Partial Hasse diagram for all substrings of p with at least two characters (thin lines) and update elements for p^+ (bold line).

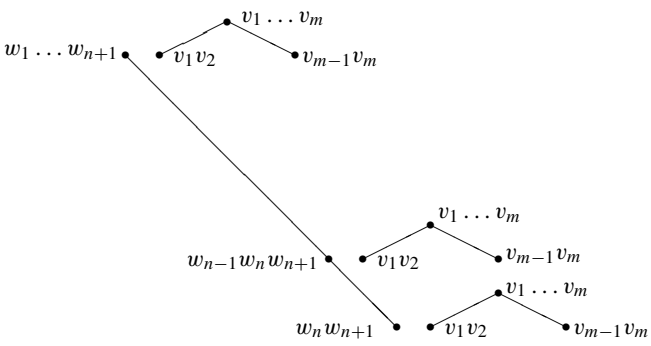

Fig. 4. Scheme for update computations for the transition from p to p^+.

Each computation requires effort $O(1)$ so that $S(x^+, y) = \max_{k=-n,\dots,m-1} S_k(x^+, y)$ can be computed with complexity $O(n + m)$ instead of the $O(nm)$ complexity of the ad-hoc procedure from Section Melting temperature and GC content, once the $O(n + m)$ intermediate values for S_k are stored.

Computing $S''(x^+, y)$ is technically more complicated than the foregoing argument since an additional character in one string may lead to a summation which—due to the 'uninterruption-condition'—did not appear for the original strings. However, the structure and complexity of the computations are similar.

Adding an initial instead of a terminal character to x and hence extending the other string y can be dealt with in a similar way. Though computing the alignment values for primers $\mathrm{sa}(p)$ and $\mathrm{sea}(p)$ still needs effort $O(n^2)$ by the resulting scheme and though each pair annealing value $\mathrm{pa}(p, q)$ and $\mathrm{pea}(p, q)$ still requires effort $O(nm)$, the computing effort for sets of primer candidates will be reduced.

Dynamic programming for enumeration

The enumeration of all self annealing values of primers $p \in \mathrm{Sub}(w)$ can be organized in a Hasse diagram which is partially sketched in Figure 3. The self annealing values of $p = w_1, \dots, w_n$ and of all its substrings are assumed to be computed and arranged below that string. Also, the intermediate values S_k from the previous section are stored. The two element strings serve for initializing the computations. The self annealing values are enumerated by the subsequent procedure.

DYNSA

1. (Initialization). Computation of $\mathrm{sa}(w_1, w_2) = S((w_2, w_1), (w_1, w_2)) = \max\{s(w_1, w_1), s(w_1, w_2) + s(w_2, w_1), s(w_2, w_2)\}$.
2. (Iteration). For $n = 2, \dots, N - 1$ do
   (a) Computation of $\mathrm{sa}(w_n, w_{n+1})$ as in the initialization.
   (b) For $i = 1, \dots, n - 1$ do
   Computation of $\mathrm{sa}(w_{n-i}, \dots, w_{n+1}) = S((w_{n+1}, \dots, w_{n-i}), (w_{n-i}, \dots, w_{n+1}))$ from values of S_k and update of these values as in Section Dynamic programming requisites.

The effort to compute the self annealing values of additional strings $(w_n, w_{n+1}), (w_{n-1}, w_n, w_{n+1}), \dots, (w_1, \dots, w_{n+1})$ is $O(1) + \dots + O(n) = O(n^2)$ according to the update scheme, see above. For $n = 2, \dots, N$ this results in effort $O(2^2) + \dots + O(N^2) = O(N^3)$ for the computation of all self annealing values of $\mathrm{Sub}(w)$. This appears to be optimal in the sense of worst case complexity since the cardinality of $\mathrm{Sub}(w)$ already is $\Theta(N^2)$. Also, the values $\mathrm{sea}(p)$ for all $p \in \mathrm{Sub}(w)$ are computed in $O(N^3)$ and the values $\mathrm{sa}(q)$ and $\mathrm{sea}(q)$ for all $q \in \mathrm{Sub}(v)$ are computed in $O(M^3)$. The procedures are minor modifications of **DYNSA**.

The enumeration of all pair annealing values $\mathrm{pa}(p, q)$ for $(p, q) \in \mathrm{Sub}(w) \times \mathrm{Sub}(v)$ proceeds by alternation. Every additional string from $\mathrm{Sub}(w)$ is combined with every string from $\mathrm{Sub}(v)$ that has been computed so far and vice versa. This is sketched for the transition from $p = w_1 \dots w_n$ to $p^+ = w_1 \dots w_{n+1}$ in Figure 4. In the subsequent algorithm the Hasse diagrams of string pairs with previously computed annealing values grow with almost same size until the smaller window is reached. Computations proceed then for the remainder of the larger window.

DYNPA

1. (Initialization). Computation of $\mathrm{pa}((w_1, w_2), (v_1, v_2)) = \max\{s(w_1, v_1), s(w_1, v_2) + s(w_2, v_1), s(w_2, v_2)\}$.

2. (Iteration). For $k = 2, \ldots, \max\{N, M\} - 1$ do
   (a) If $k \leqslant N - 1$ then for $i = 0, \ldots, k - 1$ do Computation of $\mathrm{pa}((w_{k-i}, \ldots, w_{k+1}), q)$ $\forall q \in \mathrm{Sub}(v_1, \ldots, v_{\min\{k,M\}})$.
   (b) If $k \leqslant M - 1$ then for $i = 0, \ldots, k - 1$ do Computation of $\mathrm{pa}(p, (v_{k-i}, \ldots, v_{k+1}))$ $\forall p \in \mathrm{Sub}(w_1, \ldots, w_{\min\{k+1,N\}})$.

The complexity of the update operations from p to p^+ is $n \cdot (m-1)m/2\ O(N+M)$. All updates for subsequences of w thus require $\sum_{n=2}^{N} n(m-1)m/2\ O(N+M) \leqslant (M-1)M/2\ O(N+M) \sum_{n=2}^{N} n = O((N+M)N^2M^2)$. Symmetry implies the same complexity for the updates of subsequences from window v. The overall complexity of enumerating all pair annealing values thus is $O((N+M)N^2M^2)$ which appears to be optimal since $\mathrm{Sub}(w) \times \mathrm{Sub}(v)$ already has cardinality $\Theta(N^2M^2)$. The analog holds for the enumeration of all pair end annealing values. Enumerations of these annealing values over these two windows reduces the order of complexity by one as compared with the direct approach.

Dynamic programming for approximation

Enumeration for restricted primer lengths. The length of substrings of w and v are now required to be bounded by $\lambda_w \leqslant N$ and $\lambda_v \leqslant M$ so that the actual lengths of forward and reverse primers meet the conditions $\lambda_f \leqslant \lambda_w$ and $\lambda_r \leqslant \lambda_v$. The motivations for restricting primer length are three-fold: (i) to reduce the computational effort; (ii) increase the fidelity of the PCR reaction; and (iii) lower synthesis costs.

The primer candidate sets $\mathrm{Sub}(w, \lambda_w) = \{p | p \in \mathrm{Sub}(w), |p| \leqslant \lambda_w\}$ and $\mathrm{Sub}(v, \lambda_v)$ have cardinalities $1/2\ (2N - \lambda_w)(\lambda_w - 1) = \Theta((N - \lambda_w)\lambda_w)$ and $1/2\ (2M - \lambda_v)(\lambda_v - 1) = \Theta((M - \lambda_v)\lambda_v)$ respectively. These cardinalities are in accordance with the area of a trapezoid.

The subsequent approximation enumerates the restricted primer candidate sets. The triangle sketched in Figure 3 is now cut off at height λ_w resulting in the trapezoidal subset of $\mathrm{Sub}(w)$ sketched in Figure 5.

DYNSA-APP

1. (Initialization). Computation of $\mathrm{sa}(w_1, w_2) = S((w_2, w_1), (w_1, w_2)) = \max\{s(w_1, w_1), s(w_1, w_2) + s(w_2, w_1), s(w_2, w_2)\}$.
2. (Iteration). For $n = 2, \ldots, N - 1$ do
   (a) Computation of $\mathrm{sa}(w_n, w_{n+1})$.
   (b) For $i = 1, \ldots, \min\{n - 1, \lambda_w - 2\}$ do Computation of $\mathrm{sa}(w_{n-i}, \ldots, w_{n+1}) = S((w_{n+1}, \ldots, w_{n-i}), (w_{n-i}, \ldots, w_{n+1}))$ from values of S_k and update of these values as in Section Dynamic programming requisites.

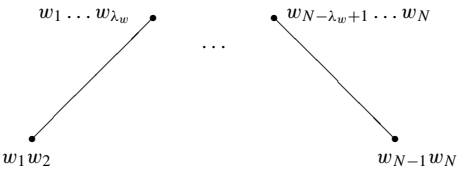

Fig. 5. Trapezoidal Hasse diagram of $\mathrm{Sub}(w, \lambda_w)$.

The complexity of each iteration of step 2 is $O(\lambda_w^2)$ resulting in an overall complexity $O(N\lambda_w^2)$ of **DYNSA-APP**. This is the complexity of computing the self annealing values of all $\Theta(N)$ substrings with exact length λ_w by the ad-hoc method. The cut off condition is algorithmically specified by the minimum condition in the inner loop.

Self-end annealing values are computed similarly to **DYNSA-APP** by a procedure called **DYNSEA-APP**. Pair annealing values for $(p, q) \in \mathrm{Sub}(w, \lambda_w) \times \mathrm{Sub}(v, \lambda_v)$ are computed by the following algorithm.

DYNPA-APP

1. (Initialization). Computation of $\mathrm{pa}((w_1, w_2), (v_1, v_2)) = \max\{s(w_1, v_1), s(w_1, v_2) + s(w_2, v_1), s(w_2, v_2)\}$.
2. (Iteration). For $k = 2, \ldots, \max\{N, M\} - 1$ do
   (a) If $k \leqslant N - 1$ then for $i = 0, \ldots, \min\{k - 1, \lambda_w - 2\}$ do Computation of $\mathrm{pa}((w_{k-i}, \ldots, w_{k+1}), q)$ $\forall q \in \mathrm{Sub}(v_1, \ldots, v_{\min\{k,M\}}, \lambda_v)$.
   (b) If $k \leqslant M - 1$ then for $i = 0, \ldots, \min\{k - 1, \lambda_v - 2\}$ do Computation of $\mathrm{pa}(p, (v_{k-i}, \ldots, v_{k+1}))$ $\forall p \in \mathrm{Sub}(w_1, \ldots, w_{\min\{k+1,N\}}, \lambda_w)$.

Each of the $O(N)$ update operations for substrings of w has a complexity bounded in $\min\{k, \lambda_w\}(\lambda_v - 1)\lambda_v/2\ O(\lambda_v + \lambda_w)$ resulting in $\sum_{k=2}^{N} \min\{k, \lambda_w\}(\lambda_v - 1)\lambda_v/2\ O(\lambda_v + \lambda_w) = O((\lambda_v + \lambda_w)\lambda_v^2 N\lambda_w)$. The complexity of **DYNPA-APP** thus is $O((\lambda_w + \lambda_v)(\lambda_v^2 N\lambda_w + \lambda_w^2 M\lambda_v)) = O((\lambda_w + \lambda_v)\lambda_w\lambda_v(\lambda_v N + \lambda_w M)) = O(\Lambda^4(N+M))$, where $\Lambda = \max\{\lambda_w, \lambda_v\}$. Computations for pair end annealing values are similar.

Approximation for bounded criteria vectors. Lengths of primers are now specified to lie within some interval which reasonably encloses the ideal values. The additional lower bounds give even more control on the design.

Primer candidates are restricted to lie in sets like $\mathrm{Sub}(w, \lambda_{l,1}, \lambda_{u,1}) = \{p | p \in \mathrm{Sub}(w), \lambda_{l,1} \leqslant |p| \leqslant \lambda_{u,1}\}$ for $\lambda_{l,1} < \lambda_{u,1}$. The size of such a set is $\Theta((\lambda_{u,1} - \lambda_{l,1})N)$. Bounding the primer length can be extended to all criteria.

For the sake of simplicity all bounds are identical for forward and reverse primer with the following notation.

$$\begin{array}{ccccc} \lambda_l & \leqslant & |p|, |q| & \leqslant & \lambda_u \\ GC_l & \leqslant & GC(p), GC(q) & \leqslant & GC_u \\ T_{m,l} & \leqslant & T_m(p), T_m(q) & \leqslant & T_{m,u} \\ 0 & \leqslant & \mathrm{sa}(p), \mathrm{sa}(q) & \leqslant & \mathrm{sa}_u \\ 0 & \leqslant & \mathrm{sea}(p), \mathrm{sea}(q) & \leqslant & \mathrm{sea}_u \\ 0 & \leqslant & \mathrm{pa}(p,q) & \leqslant & \mathrm{pa}_u \\ 0 & \leqslant & \mathrm{pea}(p,q) & \leqslant & \mathrm{pea}_u\,. \end{array}$$

Bounding allows the following algorithmic scheme which approximates a best pair of primers (cf. Spellman, 1997).

BOUND

1. Computation of the individual feasibility sets $\gamma(w) = \{p \in \mathrm{Sub}(w) | \lambda_l \leqslant |p| \leqslant \lambda_u, GC_l \leqslant GC(p) \leqslant GC_u, T_{m,l} \leqslant T_m(p) \leqslant T_{m,u}, 0 \leqslant \mathrm{sa}(p) \leqslant \mathrm{sa}_u, 0 \leqslant \mathrm{sea}(p) \leqslant \mathrm{sea}_u\}$ and $\delta(v) = \{q \in \mathrm{Sub}(v) | \lambda_l \leqslant |q| \leqslant \lambda_u, GC_l \leqslant GC(q) \leqslant GC_u, T_{m,l} \leqslant T_m(q) \leqslant T_{m,u}, 0 \leqslant \mathrm{sa}(q) \leqslant \mathrm{sa}_u, 0 \leqslant \mathrm{sea}(q) \leqslant \mathrm{sea}_u\}$.
2. Computation of conjoint feasibility set $\varepsilon(w,v) = \{(p,q) \in \gamma(w) \times \delta(v) | \ 0 \leqslant \mathrm{pa}(p,q) \leqslant \mathrm{pa}_u, 0 \leqslant \mathrm{pea}(p,q) \leqslant \mathrm{pea}_u\}$.
3. Computation of best primer pair $(p^1, q^1) = \mathrm{argmin}_{(p,q)\in\varepsilon(w,v)} || \mathrm{sc}(p,q) - \mathrm{sc}_{\mathrm{ideal}} ||_\kappa$.

In case one of the individual feasibility sets is empty, the bounds on criteria involving a single primer are to be relaxed. In case the conjoint feasibility set is empty, bounds on pair annealing or pair end annealing are to be relaxed. The computationally expensive consideration of primer pairs is deferred by the last algorithm until step 2, where primers which violate the individual bounds have been eliminated. Step 1 can be handled by algorithms **DYNSA-APP** and **DYNSEA-APP** in $O((N+M)\lambda_u^2)$. Step 2 takes effort $O(|\gamma(w)||\delta(v)|\lambda_u^2)$ if ad-hoc computations are applied. This is reasonable when the bounds from step 1 effectively reduce the product set $\gamma(w) \times \delta(v)$ to the conjoint feasibility set from step 2. Moreover, monotonicity of the pair annealing values $\mathrm{pa}(p,q)$ can be exploited when reducing the product set $\gamma(w) \times \delta(v)$ to the conjoint feasibility set $\varepsilon(w,v)$. Step 3 obviously requires only the effort $O(|\varepsilon(w,v)|)$.

MULTIPLEX PRIMER ASSESSMENT

Multiplex PCR is assumed here to operate in a single wet container, where μ DNA targets are to be amplified simultaneously. The intervals specified by window pairs for forward and reverse primers are pairwise disjoint, (see Figure 6). The windows $w^i = (w^i_1, \ldots, w^i_{N_i})$ and $v^i = (v^i_1, \ldots, v^i_{M_i})$ may have different lengths.

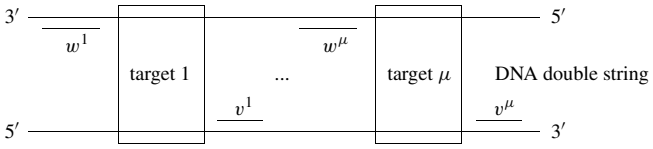

Fig. 6. Multiple target situation.

The task is to find a multiplex primer tuple $\mathcal{M} = ((p^1, q^1), \ldots, (p^\mu, q^\mu))$ with $p^i \in \mathrm{Sub}(w^i)$ and $q^i \in \mathrm{Sub}(v^i)$ for $i = 1, \ldots, \mu$. The tentative scoring vector of the multiplex primer tuple is simply defined be the joint vector of the scores of all primer pairs

$$\mathrm{sc}(\mathcal{M})^{ten} = (\mathrm{sc}(p^1, q^1), \ldots, \mathrm{sc}(p^\mu, q^\mu))^T \in \mathbb{R}^{12\mu}.$$

The corresponding, tentative ideal scoring vector is formed of the μ individually ideal scoring vectors $\mathrm{sc}_{1,\mathrm{ideal}}, \ldots, \mathrm{sc}_{\mu,\mathrm{ideal}}$ by

$$\begin{aligned} \mathrm{sc}^{\mu,\mathrm{ten}}_{\mathrm{ideal}} &= (\mathrm{sc}_{1,\mathrm{ideal}}, \ldots, \mathrm{sc}_{\mu,\mathrm{ideal}}) \\ &= (\mathrm{length}_{1,f}, \mathrm{length}_{1,r}, GC_{1,f}, GC_{1,r}, T_{m,1,f}, \\ &\quad T_{m,1,r}, 0, 0, 0, 0, 0, 0, \ldots, \\ &\quad \mathrm{length}_{\mu,f}, \mathrm{length}_{\mu,r}, GC_{\mu,f}, GC_{\mu,r}, \\ &\quad T_{m,\mu,f}, T_{m,\mu,r}, 0, 0, 0, 0, 0, 0)^{\mathrm{T}}. \end{aligned}$$

Uniform physical conditions of multiplex PCR leads to choose all ideal temperatures to be equal $T_m = T_{m,1,f} = T_{m,1,r} = \cdots = T_{m,\mu,f} = T_{m,\mu,r}$. Pair annealing and pair end annealing between all primers from different windows is only taken into consideration by the final scoring vector

$$\begin{aligned} \mathrm{sc}(\mathcal{M}) = (&\mathrm{sc}(p^1, q^1), \ldots, \mathrm{sc}(p^\mu, q^\mu), \\ &\mathrm{pa}(p^1, p^2), \mathrm{pea}(p^1, p^2), \mathrm{pa}(p^1, p^3), \mathrm{pea}(p^1, p^3) \\ &, \ldots, \mathrm{pa}(p^{\mu-1}, p^\mu), \mathrm{pea}(p^{\mu-1}, p^\mu), \\ &\mathrm{pa}(p^1, q^2), \mathrm{pea}(p^1, q^2), \mathrm{pa}(p^1, q^3), \mathrm{pea}(p^1, q^3) \\ &, \ldots, \mathrm{pa}(p^{\mu-1}, q^\mu), \mathrm{pea}(p^{\mu-1}, q^\mu), \\ &\mathrm{pa}(p^2, q^1), \mathrm{pea}(p^2, q^1), \mathrm{pa}(p^3, q^1), \mathrm{pea}(p^3, q^1) \\ &, \ldots, \mathrm{pa}(p^\mu, q^{\mu-1}), \mathrm{pea}(p^\mu, q^{\mu-1}), \\ &\mathrm{pa}(q^1, q^2), \mathrm{pea}(q^1, q^2), \mathrm{pa}(q^1, q^3), \mathrm{pea}(q^1, q^3) \\ &, \ldots, \mathrm{pa}(q^{\mu-1}, q^\mu), \mathrm{pea}(q^{\mu-1}, q^\mu))^T \\ &\in \mathbb{R}^{12\mu+4\mu(\mu-1)} = \mathbb{R}^{4\mu^2+8\mu}. \end{aligned}$$

Symmetry of both pair annealing and pair-end annealing functions ensures that each primer is assessed in combination with each other. The corresponding ideal scoring

vector is obtained by inserting the coordinate 0 for every additional pair and pair end annealing value

$$\mathrm{sc}^{\mu}_{\mathrm{ideal}} = (\mathrm{sc}^{\mu,\mathrm{ten}}_{\mathrm{ideal}}, 0, \ldots, 0)^{\mathrm{T}}.$$

Thus, the formal objective of multiplex PCR can be stated as:

$$\min_{\mathcal{M}\in \mathrm{Sub}(w^1)\times \mathrm{Sub}(v^1)\times\cdots\times \mathrm{Sub}(w^\mu)\times \mathrm{Sub}(v^\mu)} ||\mathrm{sc}(\mathcal{M}) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}||_\kappa,$$

where $||\,||_\kappa$ is the l_1-norn in $\mathbb{R}^{4\mu^2+8\mu}$. Weights for deviations from ideal values are the same as for ordinary PCR, see Section Aggregation of assessments for ordinary PCR. Noteworthy, the weights for pair annealing values of primers from the same window are identical to those from different windows. The same applies to pair end annealing values.

MULTIPLEX COMPUTATIONS

The set of all multiplex primer tuples has cardinality $|\mathrm{Sub}(w^1) \times \cdots \times \mathrm{Sub}(v^\mu)| = N_1(N_1 - 1)/2 \cdot \cdots \cdot M_\mu(M_\mu - 1)/2 = O((\omega^4/4)^\mu)$ with $\omega = \max\{N_1, \ldots, N_\mu, M_1, \ldots, M_\mu\}$ and hence grows exponentially in the number of targets, since $\omega^4/4 > 1$. Thus, only approximation rather than complete enumeration is investigated.

All criteria involving only a single primer or primers from one window are supposed to be bounded as in Section Approximation for bounded criteria vectors. The bounds may depend on the window pairs. This allows the computation all sets $\varepsilon(w^1, v^1), \ldots, \varepsilon(w^\mu, v^\mu)$ independently from all others in the first two steps of algorithm **BOUND**. All of the primer pair sets are supposed to be non-void.

The construction of a multiplex primer tuple may then proceed by sorting all the pair sets in order of increasing distances so that $\varepsilon(w^i, v^i) = \{(p^i_j, q^i_j) | \,||\,\mathrm{sc}(p^i_j, q^i_j) - \mathrm{sc}_{i,\mathrm{ideal}}\,||_\kappa \uparrow_j\}$. The subscripts refer to primer pairs so that identical primers showing up in different pairs may receive different indices. The individual orders of the pair sets provide a search direction for multiplex tuples. Candidates for the multiplex tuples are formed iteratively from the current best primer tuple $\mathcal{M} = ((p^1_{j_1}, q^1_{j_1}), \ldots, (p^\mu_{j_\mu}, q^\mu_{j_\mu}))$. If $\mathcal{M}$ is found to be acceptable, then computations terminate. Otherwise, a primer pair $(p^i_{j_i}, q^i_{j_i})$ is selected to be replaced by the next pair $(p^i_{j_i+1}, q^i_{j_i+1})$ resulting in the new candidate tuple $\mathcal{M}^i$. The set of all these candidates is the successor set $\mathrm{succ}(\mathcal{M})$. Selection from the successor set can be based on a variety of criteria such as leading to a new tuple with $\min_{\mathcal{M}^i} ||\,\mathrm{sc}(\mathcal{M}^i) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa$.

The procedure terminates unsuccessfully, if $\mathcal{M}$ neither is accepted nor has a successor. Acceptance of $\mathcal{M}$ can be based on the formal muliplex objective or on other criteria such as satisfaction of bounds on annealing scores of primers from different windows or on a combination of all these. Although the procedure outlined here considers primers as a function of distance to individual ideal scoring vectors, the overall scores $||\,\mathrm{sc}(\mathcal{M}) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa$ need not be encountered in a monotone fashion. Thus, the record is kept separately. This procedure is formally stated as follows.

MULT

1. Computation of primer pair sets $\varepsilon(w^1, v^1), \ldots, \varepsilon(w^\mu, v^\mu)$ by **BOUND** and sorting each set according to $||\,\mathrm{sc}(p^i_j, q^i_j) - \mathrm{sc}_{i,\mathrm{ideal}}\,||_\kappa \uparrow_j$.
2. Set $\mathcal{M}_0 = \mathcal{M} = ((p^1_1, q^1_1), \ldots, (p^\mu_1, q^\mu_1))$, set acceptance level $\varepsilon > 0$.
3. Repeat until $||\,\mathrm{sc}(\mathcal{M}) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa \leqslant \varepsilon$ or some other stopping criterion is met or until $\mathrm{succ}(\mathcal{M}) = \emptyset$: Computation of all $\mathcal{M}^i \in \mathrm{succ}(\mathcal{M})$, set $\mathcal{M} = \mathrm{argmin}_{\mathcal{M}^i} ||\,\mathrm{sc}(\mathcal{M}^i) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa$. If $||\,\mathrm{sc}(\mathcal{M}) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa < ||\,\mathrm{sc}(\mathcal{M}_0) - \mathrm{sc}^{\mu}_{\mathrm{ideal}}\,||_\kappa$, then $\mathcal{M}_0 = \mathcal{M}$.
4. Report of best multiplex tuple $\mathcal{M}_0$ found.

Every iteration of step 3 arbitrates between $O(\mu)$ tuples each being evaluated by $O(\mu^2)$ criteria of computing complexity $O(\omega^2)$ if ad-hoc computations are used. This results in a computation bound of $O(\omega^2\mu^3)$ for each iteration of step 3.

COMPUTATIONAL FINDINGS

Implementation

While the criteria based on melting temperature and GC content are straightforward from the computational viewpoint, the benefit of dynamic programming is elaborated for computations of self annealing values and self end annealing values. The dynamic expansions are therefore tuned so that updating requires only one intermediate step instead of two, as compared with the original transitions from $S(x, y)$ to $S(x^+, y)$ to $S(x^+, y^+)$ according to Section PCR primer computation. Implementational details and computational results are sketched below.

The update scheme for self annealing values is modified to

$$S_k(\tilde{x^+}, x^+) = \begin{cases} S_{k+1}(\tilde{x}, x), \\ \qquad \text{for } k = -1, \ldots, -n \\ S_{k+1}(\tilde{x}, x) + 2s(x_{k+1}, x_{n+1}), \\ \qquad \text{for } k = 0, \ldots, n-2 \\ 2s(x_{k+1}, x_{n+1}), \\ \qquad \text{for } k = n-1 \\ s(x_{k+1}, x_{n+1}), \\ \qquad \text{for } k = n \end{cases}$$

where $x = (x_1, \ldots, x_n)$, $\tilde{x} = (x_n, \ldots, x_1)$, $x^+ = (x_1, \ldots, x_{n+1})$, and $\mathrm{sa}(x^+) = \max_{k=-n,\ldots,n} S_k(\tilde{x^+}, x^+)$. A similar update formula applies to $\mathrm{sa}(x_+)$ where $x_+ = (x_0, \ldots, x_n)$. The computational update scheme for all self annealing values of primer candidates from a given string $w_1, \ldots, w_N$, (cf. Figure 3), thus results in the final iteration scheme

$$S_k(w_{n-i}, \ldots, w_{n+1}) = \begin{cases} S_{k+1}(w_{n-i+1}, \ldots, w_{n+1}), \\ \qquad \text{for } k = -1, \ldots, -(i+1) \\ S_0(w_{n-i+1}, \ldots, w_n) + 2s(w_{n-i}, w_{n+1}), \\ \qquad \text{for } k = 0 \\ S_{k-1}(w_{n-i}, \ldots, w_n), \\ \qquad \text{for } k = 1, \ldots, i+1, \end{cases}$$

where $i = 1, \ldots, n-1$ in case of unbounded sequences and $i = 1, \ldots, \lambda_w - 1$ in case of bounded primer lengths. The index n attains values $n = 2, \ldots, N$ in both cases. For $i = 0$ the three initial values are explicitly computed as $S_{-1}(w_n, w_{n+1}) = s(w_n, w_n)$, $S_0(w_n, w_{n+1}) = 2s(w_n, w_{n+1})$, and $S_1(w_n, w_{n+1}) = s(w_{n+1}, w_{n+1})$. Then

$$\mathrm{sa}(w_{n-i}, \ldots, w_{n+1}) = \max_{k=-(i+1),\ldots,i+1} S_k(w_{n-i}, \ldots, w_{n+1}).$$

Only a single addition needs to be made for each pair of values i and n.

Implementations of procedures **DYNSA** and **DYNSA-APP** are based on the final iteration scheme. An analogous scheme applies to the computation of self end annealing values providing the base for efficient implementations of **DYNSEA** and **DYNSEA-APP**.

Pair annealing values are computed similarly along so-called central alignments. These allow to express the scoring value for $w = (w_1, \ldots, w_n)$ and $v = (v_1, \ldots, v_m)$ by

$$\begin{aligned} \mathrm{pa}(w, v) &= \max_{k=-(n-1),\ldots,m-1} S_k(\tilde{w}, v) \\ &= \max_{k=-(n-1),\ldots,m-1} S_0(\tilde{w(k)}, v(k)), \end{aligned}$$

where the substrings $w(k)$ of w and $v(k)$ of v denote the proper overlay for shift k. The proper overlay is given for $-(n-1) \leqslant k \leqslant 0$ by

$$\begin{array}{ccccccc} v_1 & \ldots & v_{n+k} & \ldots & v_m \\ w_{n+k} & \ldots & w_1 \end{array}$$

or

$$\begin{array}{ccccc} v_1 & \ldots & v_m \\ w_{n+k} & \ldots & w_{n+k-m+1} & \ldots & w_1 \end{array}$$

so that $S_0(\tilde{w(k)}, v(k)) = S_0((w_{n+k}, \ldots, w_1), (v_1, \ldots, v_{n+k}))$ in the first case which is $n + k \geqslant m$ and $S_0(\tilde{w(k)}, v(k)) = S_0((w_{n+k}, \ldots, w_{n+k-m+1}), (v_1, \ldots, v_m))$ in the second case which is $n + k < m$. For $0 \leqslant k \leqslant m-1$ the proper overlays are given by

$$\begin{array}{ccccccc} v_1 & \ldots & v_{1+k} & \ldots & v_m \\ & & w_n & \ldots & w_{n-m+k+1} & \ldots & w_1 \end{array}$$

or

$$\begin{array}{ccccccc} v_1 & \ldots & v_{k+1} & \ldots & v_{k+n} & \ldots & v_m \\ & & w_n & \ldots & w_1 \end{array}$$

This leads to $S_0(\tilde{w(k)}, v(k)) = S_0((w_n, \ldots, w_{n-m+k+1}), (v_{1+k}, \ldots, v_m))$ in the first case which is $k + n > m$ and it leads to $S_0(\tilde{w(k)}, v(k)) = S_0((w_n, \ldots, w_1), (v_{k+1}, \ldots, v_{k+n}))$ in the second case which is $k + n \leqslant m$. This makes it possible to compute scores for central alignments efficiently and then compute the annealing score for each primer pair in time proportional to their common length. Formally, this results in the following procedure considering the upper bounds λ_w and λ_v on the lengths of the forward and reverse primers respectively.

DYNPA-APP2

1. Computation of

$$\begin{aligned} &S_0((w_{i+l-1}, \ldots, w_i), (v_j, \ldots, v_{j+l-1})) \\ &= S_0((w_{i+l-1}, \ldots, w_{i+1}), (v_j, \ldots, v_{j+l-2})) \\ &+ s(w_i, v_{j+l-1}) \end{aligned}$$

   for all $l = 2, \ldots, \lambda$, $i = 1, \ldots, N - l$, and $j = 1, \ldots, M - l$. The value λ is the maximum possible length of the central alignments which is $\lambda = \min\{\lambda_w, \lambda_v\}$.

2. Computation of pair annealing values for each pair $w = (w_1, \ldots, w_n)$ and $v = (v_1, \ldots, v_m)$ via

$$\begin{aligned} \mathrm{pa}(w, v) &= \max_{k=-(n-1),\ldots,m-1} S_k(\tilde{w}, v) \\ &= \max_{k=-(n-1),\ldots,m-1} S_0(\tilde{w(k)}, v(k)). \end{aligned}$$

Step 1 requires $O(\lambda NM)$ time and space. The computations for each pair in step 2 require time $O(n + m) = O(\lambda_v + \lambda_w)$ whenever S_0 values can be accessed in $O(1)$. The complexity of computing all pair annealing values is then $O(|\,\mathrm{Sub}(w)| \cdot |\,\mathrm{Sub}(v)| \cdot (\lambda_v + \lambda_w)) + O(\lambda NM) = O(N\lambda_w \cdot M\lambda_v \cdot (\lambda_v + \lambda_w)) + O(\lambda NM) = O(\Lambda^3 NM)$.

The **BOUND** procedure was implemented in the system DOPRIMER (Design of oligonucleotide primers) which makes use of the procedure **DYNSA-APP** and of the related procedure **DYNSEA-APP**. The following environment was used.

Table 1. Computation times for self annealing values and self end annealing values.

Primer length $\lambda_l \ldots \lambda_u$	18 . . . 21			15 . . . 26		
Window length N	35	300	1000	35	300	1000
No. of primers $\lvert \mathrm{Sub}(w, \lambda_l, \lambda_u) \rvert$	66	1126	3926	186	3366	11766
Straightforward computations of sa [in ms]	271	4631	16162	800	15380	53938
computations by **DYNSA-APP** [in ms]	4	62	209	6	84	292
Relative savings	**67.75**	**74.7**	**77.3**	**133.3**	**183.1**	**184.7**
Straightforward computations of sea [in ms]	27	440	1550	80	1371	4852
computations by **DYNSEA-APP** [in ms]	4	45	156	4	64	224
Relative savings	**6.75**	**9.7**	**9.9**	**20**	**21.4**	**21.6**

- Pentium 200 MHz, 32 MB,
- Linux 2.0.32 (RedHat 5.0),
- jdk-1.1.5 (JAVA development kid without JIT).

A couple of measures of code optimization such as the use of data structures allowing fast access under JAVA were taken but no details are reported here.

Numerical results

Extensive studies were performed with parameters stated in the subsequent table. The values refer to computations by procedures **DYNSA-APP** and **DYNSEA-APP**. The entry 'no. of primers' denotes the number of all primers of restricted lengths each within a window of specified length. Run times are averages over sample sets consisting of at least one hundred elements. However, variations in run times turned out to be negligible as could have been anticipated from the construction of the algorithms in Table 1.

Comparison of the measured computing times reveals that the dynamic programming scheme increases the calculation speed by at least 50 : 1 in the case of the given annealing values. The gain in computing time for self end annealing is less but still significant. The relative savings ratios increase as a function of primer length and window length. Window lengths of up to 300 and primer lengths of up to 26 are realistic both for ordinary and multiplex PCR. Window lengths as large as 1000 are relevant to hybridization on arrays of oligonucleotides which will be dealt with elsewhere.

CONCLUSION

Dynamic programming has been applied for calculating oligonucleotide interactions for PCR primer pairs. The scheme resulted in a relative gain in computing speed of up to 50 : 1 on a given data set. The increases which dynamic programming provides indicates that the full potential of mathematical calculation tools for this type of calculation has as yet to be realized.

The ever increasing size of nucleic acid databases combined with the continued development of nucleic acid arrays will require the implementation of ever more rapid calculation strategies. This approach has obvious implications for other primer based amplification techniques. Furthermore, the increased application of nucleic acids for 'nonbiological' purposes ranging from the construction of nanostructures to their use as tags to direct the positioning of molecules will benefit greatly from these efforts.

The algorithms currently employed require the use of correction factors in order to obtain T_ms which correspond to empirically determined values. Clearly important factors which influence the stability of nucleic acids have yet to be identified. It should be possible to systematically correlate the calculated T_ms of primers with their empirically determined values to derive better correction factors and thereby improve the predicative ability of these algorithms. And finally, the increased throughput of this scheme will allow the systematic comparison of the chemical and physical properties of primers providing important clues essential in the effort to delineate a complete structure–function map of this important class of biomolecules.

ACKNOWLEDGEMENTS

We would like to thank F.Ortigao and C.Sarkar (both of INTERACTIVA Biotechnologie GmbH, Ulm) for explaining to us some of the magic of PCR. The work was supported by the 'Design of Oligonucleotide Primers' grant to INTERACTIVA from the program 'Zukunftsinitiative Junge Generation', Baden-Württemberg, Germany and an 'Exploratory Research' grant to Michael Mecklenburg from the National Industrial and Technical Board (NUTEK), Sweden.

REFERENCES

Apostolico,A. and Galil,Z. (1997) *Pattern Matching Algorithms*. Oxford University Press, Oxford.

Borer,P.N. *et al.* (1974) Stability of ribonucleic acid double-stranded helices. *J. Mol. Biol.*, **86**, 843–853.

Breslauer,K.J. *et al.* (1986) Predicting DNA duplex stability from the base sequence. *Proc. Natl Acad. Sci. USA*, **83**, 3746–3750.

Edwards,M.C. and Gibbs,R.A. (1994) Multiplex PCR: advantages, development and applications. *PCR Meth. Appl.*, **3**, 565–575.

Freier,S.M. *et al.* (1986) Improved free-energy parameters for predictions of RNA duplex stability. *Proc. Natl Acad. Sci. USA*, **83**, 9373–9377.

Hillier,L. and Green,P. (1991) OSP: a computer program for predictions of DNA duplex stability. *PCR Meth. Appl.*, **1**, 124–128.

Keeney,R.L. and Raiffa,H. (1976) *Decisions with Multiple Objectives*. Wiley, New York.

Mecklenburg,M. (1997) Design of high-annealing-temperature primers for PCR and development of a versatile low-copy-number amplification protocol. *Adv. Mol. Cell Biol.*, **15**B, 473–490.

Owczarzy,R. *et al.* (1998) Predicting sequence-dependent melting stability of short duplex DNA oligomers. *Biopoly*, **44**, 217–239.

Rychlik,W. and Rhoads,R.E. (1989) A computer program for choosing optimal oligonucleotides for filter hybridization, sequencing an in vitro amplification of DNA. *Nucleic Acids Res.*, **19**, 8543–8551.

Rychlik,W., Spencer,W.J. and Rhoads,R.E. (1990) Optimization of the annealing temperature for DNA amplification in vitro. *Nucleic Acids Res.*, **18**, 6409–6412.

Rychlik,W. (1995) Selection of primers for polymerase chain reaction. *Mol. Biotechnol.*, **3**, 129–134.

Spellman,P. (1997) Web primer software manual. http://genome-www2.stanford.edu/cgi-bin/SGD/webprimer.

Waterman,M.S. (1984) Efficient sequence alignment algorithms. *J. Theor. Biol.*, **108**, 333–337.

Yu,P.L. (1989) Multiple criteria decision making: five basic concepts. In Nemhauser,G., *et al.* (eds), *Handbooks in or and Management Science I,* Optimization, North-Holland, Amsterdam, pp. 663–699.