Universität Bielefeld
Technische Fakultät
AG Genominformatik

Microarray Layout and the Quadratic Assignment Problem

Sérgio A. de Carvalho Jr.[1,2,3] **and Sven Rahmann**[2,3]

[1] Graduiertenkolleg Bioinformatik, [2] International NRW Graduate School in Bioinformatics and Genome Research,
[3] Algorithms and Statistics for Systems Biology, Genome Informatics, Technische Fakultät, Universität Bielefeld, D-33594 Bielefeld, Germany.

1. Introduction

Oligonucleotide microarrays consist of small DNA fragments (probes) chemically synthesized at specific locations (spots) of a solid surface. Probes are typically 25-60 nucleotides long and are synthesized in parallel, in a series of repetitive steps. Each step appends a particular nucleotide to selected regions of the chip. Selection occurs by exposure to light with the help of a photolithographic mask or micromirror arrays.

Formally, we have a set of probes $\mathcal{P} = \{p_1, p_2, ...p_n\}$ that are produced by a series of masks $\mathcal{M} = (m_1, m_2, ...m_T)$, where each mask m_t induces the addition of a particular nucleotide $\mathcal{S}_t \in \{A, C, G, T\}$ to a subset of $\mathcal{P}$. The sequence of nucleotides added in each step $\mathcal{S} = \mathcal{S}_1\mathcal{S}_2 \ldots \mathcal{S}_T$ is called deposition sequence.

In general, a probe can be embedded within $\mathcal{S}$ in several ways. An embedding of p_k is a T-tuple $\varepsilon_k = (e_{k,1}, e_{k,2}, ...e_{k,T})$ in which $e_{k,t} = 1$ if probe p_k receives nucleotide $\mathcal{S}_t$, or 0 otherwise.

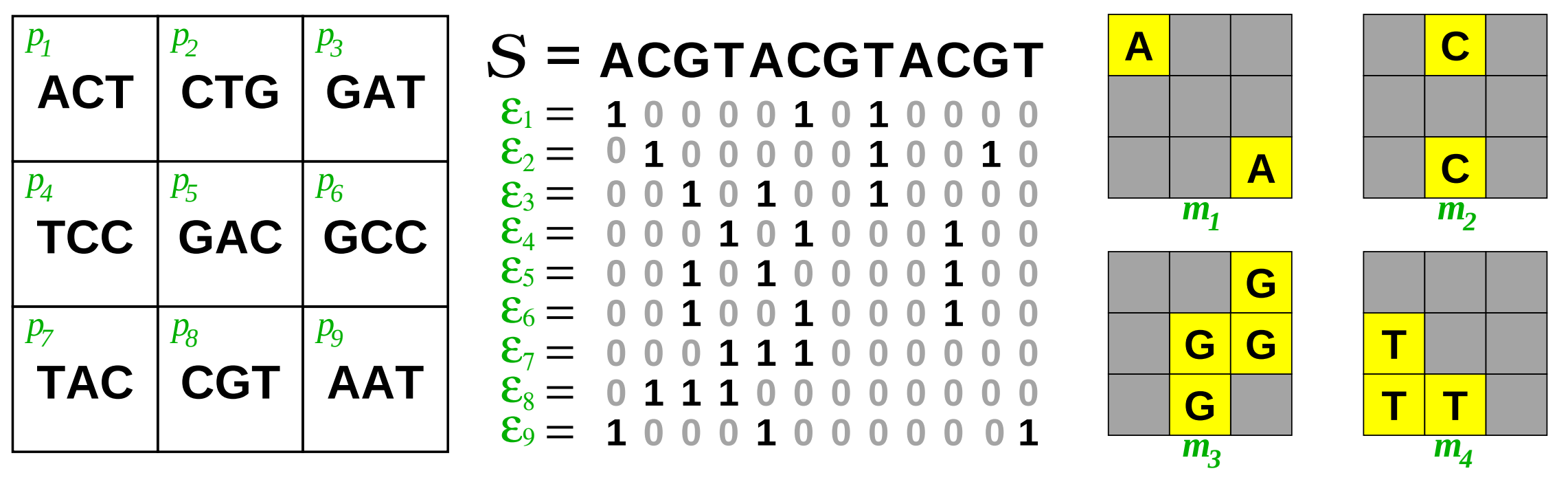

p_1 ACT
p_2 CTG
p_3 GAT
p_4 TCC
p_5 GAC
p_6 GCC
p_7 TAC
p_8 CGT
p_9 AAT
S = ACGTACGTACGT
ε_1 = 1 0 0 0 0 1 0 1 0 0 0 0
ε_2 = 0 1 0 0 0 0 0 1 0 0 1 0
ε_3 = 0 0 1 0 1 0 0 1 0 0 0 0
ε_4 = 0 0 0 1 0 1 0 0 0 1 0 0
ε_5 = 0 0 1 0 1 0 0 0 0 1 0 0
ε_6 = 0 0 1 0 0 1 0 0 0 1 0 0
ε_7 = 0 0 0 1 1 1 0 0 0 0 0 0
ε_8 = 0 1 1 1 0 0 0 0 0 0 0 0
ε_9 = 1 0 0 0 1 0 0 0 0 0 0 1
A
A
m_1
C
C
m_2
G
G G
G
m_3
T
T T
m_4

Figure 1: Synthesis of a hypothetical 3×3 chip in 12 steps. Left: chip layout and 3-nt-long probe sequences. Center: deposition sequence and probe embeddings. Right: first four resulting masks

Due to diffraction of light or internal reflection, untargeted probes can be accidentally activated. This is more likely to occur near the borders between masked and unmasked spots – hence the term border conflict.

Border Length

The Border Length Minimization Problem [1] aims at finding an arrangement of the probes together with their embeddings with minimum border conflicts.

The border length $\mathcal{B}_t$ of a mask m_t is defined as the number of borders separating masked and unmasked spots at synthesis step t. The total border length of an arrangement is the sum of border lengths of all masks. The four masks shown in Figure 1 have $\mathcal{B}_1 = 4$, $\mathcal{B}_2 = 6$, $\mathcal{B}_3 = 6$ and $\mathcal{B}_4 = 4$. The total border length of that arrangement is 50 (masks m_5 to m_12 not shown).

2. Conflict Index

We are more interested in estimating the risk of synthesizing a particular probe incorrectly. Additionally, we want to take into account that:

a) imperfections produced in the middle of a probe are more harmful than in its extremities,

b) stray light might activate probes that lie as far as three cells away from the targeted spot.

This motivates the following definition of the conflict index $\mathcal{C}(p)$ of a probe p:

$$\mathcal{C}(p) := \sum_{t=1}^{T}\Big(\omega(p,t)\sum_{p'}\delta(p,p',t)\Big),$$

where $\omega(p,t)$ are position-dependent weights (observation a) and $\delta(p,p',t)$ are distance-dependent weights (observation b) defined as follows.

Position and Distance-dependent Weights

$$\omega(p,t) := \begin{cases} c \cdot \exp\left(\theta \cdot \lambda(p,t)\right) & \text{if } p \text{ is masked at step } t, \\ 0 & \text{otherwise,} \end{cases}$$

where $c > 0$ and $\theta > 0$ are constants,

$$\lambda(p,t) := 1 + \min(b_{p,t}, \ell_p - b_{p,t}),$$

$b_{p,t}$ denotes the number of nucleotides synthesized within p up to and including step t, and ℓ_p is the length of probe p. We set $\theta := 5/\ell_p$ and $c := 1/\exp(\theta)$; see Figure 2, left.

$$\delta(p,p',t) := \begin{cases} (d(p,p'))^{-2} & \text{if } p' \text{ is unmasked at step } t, \\ 0 & \text{otherwise,} \end{cases}$$

where $d(p,p')$ is the Euclidean distance between the spots of p and p'. We restrict the support of $\delta(p,p',\cdot)$ to those $p' \neq p$ inside a 7×7 grid centered on p (Figure 2, right).

3. Quadratic Assignment Problem (QAP)

The QAP is a classical combinatorial optimization problem that can be stated as follows. Given $n \times n$ real-valued matrices $F = (f_{ij}) \geq 0$ and $D = (d_{kl}) \geq 0$, find a permutation π of $\{1, 2, \ldots n\}$ such that

$$\sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij} \cdot d_{\pi(i)\pi(j)} \rightarrow \min.$$

One of its applications is to model the Facility Location Problem where n facilities must be assigned to n locations: F is called flow matrix as f_{ij} represents the flow of materials from facility i to facility j; D is called distance matrix as d_{kl} gives the distance between locations k and l. Permutation π defines an assignment of facilities to locations with minimum cost.

QAP Formulation of Placement Problem

The probe placement problem is an instance of the QAP [2]: we want to find a one-to-one correspondence between probes and spots minimizing the total border length or sum of conflict indices.

All probes are assumed to have a single pre-defined embedding in order to force a one-to-one relationship. When there are more spots than probes, we add "empty" probes and define their weights appropriately.

For conflict index minimization, the flow f_{ij} between spots i and j is:

$$f_{ij} := \begin{cases} (d(i,j))^{-2} & \text{if spot } j \text{ is near spot } i, \\ 0 & \text{otherwise.} \end{cases}$$

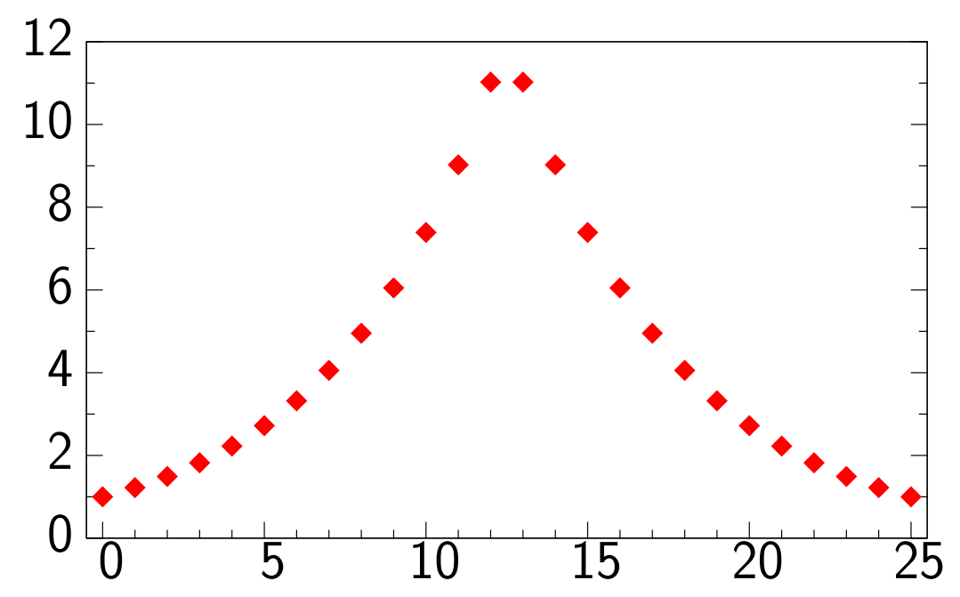
where "near" means that spot j is at most three cells away from i. The distance d_{kl} between probes k and l is:

$$d_{kl} := \sum_{t=1}^{T} d_{klt},$$

where d_{klt} is the potential contribution of probe l's embedding to the failure risk of probe p_k in the t-th synthesis step:

$$d_{klt} = \begin{cases} c \cdot \exp(\theta \cdot \lambda(p_k,t)) & \text{if } p_k \text{ is masked and } p_l \text{ unmasked} \\ & \text{in step } t, \\ 0 & \text{otherwise.} \end{cases}$$

For border length minimization we set $f_{ij} := 1$ if spots i and j are direct neighbors, $f_{ij} := 0$ otherwise; $\theta = 0$ and $c = 1/2$.



0.06	0.08	0.10	0.11	0.10	0.08	0.06
0.08	0.13	0.20	0.25	0.20	0.13	0.08
0.10	0.20	0.50	1.00	0.50	0.20	0.10
0.11	0.25	1.00	p	1.00	0.25	0.11
0.10	0.20	0.50	1.00	0.50	0.20	0.10
0.08	0.13	0.20	0.25	0.20	0.13	0.08
0.06	0.08	0.10	0.11	0.10	0.08	0.06

Figure 2: Values of ω and δ for a typical Affymetrix chip with 25-nt-long probes. Left: values of $\omega(p,t)$ on the y-axis for each value of $b_{p,t}$ on the x-axis, assuming that p is masked at step t. Right: approximate values of $\delta(p,p',t)$ for a probe p in the center and neighbors p', assuming that p' is unmasked.

4. Results and Outlook

The QAP is known to be NP-hard and particularly hard to solve in practice. Instances of size larger than $n = 20$ are generally considered to be impossible to solve to optimality. Nevertheless, our formulation is of interest because existing QAP heuristics can now be used to solve the placement problem.

We used GRASP with Path-Relinking [3] to design small artificial microarrays, and compared them with the layouts produced by Row-epitaxial [4], the best known placement algorithm (Table 1).

Table 1: Total border length and average conflict index of chips produced by Row-epitaxial and GRASP with Path-Relinking (reported times in seconds).

	Border length minimization				Conflict index minimization			
	Row-epitaxial		GRASP-PR		Row-epitaxial		GRASP-PR	
Chip size	B. length	Time	B. length	Time	C. index	Time	C. index	Time
6×6	1 714.60	0.01	1 672.20	2.73	495.15	0.05	467.08	3.68
7×7	2 354.60	0.02	2 332.60	6.43	521.90	0.07	489.32	8.84
8×8	3 123.80	0.03	3 099.13	12.49	551.84	0.09	515.69	19.48
9×9	3 974.80	0.05	3 967.20	25.96	568.62	0.11	533.79	38.83
10×10	4 895.60	0.06	4 911.40	47.57	576.49	0.11	539.69	73.09
11×11	5 954.40	0.10	5 990.73	87.48	588.91	0.12	551.41	145.67
12×12	7 086.20	0.11	7 159.80	152.42	598.21	0.12	561.21	249.19

Because of the large number of probes on industrial microarrays, it is not feasible to use a QAP algorithm to design an entire microarray, but we showed that it is possible to use it on small sub-regions of a chip. We see two applications for this approach:

1) combine it with a *partitioning algorithm* that divides the placement problem into smaller sub-problems;

2) use it to improve an existing layout, iteratively, by relocating probes inside a *sliding-window* (with small changes to the QAP algorithm).

Several QAP instances derived with our formulations are available for further investigation of other QAP algorithms at:

http://gi.cebitec.uni-bielefeld.de/assb/chiplayout/qap

References

[1] Hannenhalli, S., Hubell, E., Lipshutz, R., Pevzner, P. (2002): Combinatorial algorithms for design of DNA arrays. Adv. Biochem. Eng. Biotechnol. **77**, 1–19.

[2] de Carvalho Jr., S., Rahmann, S. (2006): Microarray Layout as a Quadratic Assignment Problem. In *German Conference on Bioinformatics (GCB)*, LNI, Springer. To appear.

[3] Oliveira,C.A.S., Pardalos,P.M. and Resende,M.G.C. (2004): GRASP with path-relinking for the quadratic assignment problem. In *Efficient and Experimental Algorithms*, LNCS **3059**, 356–368, Springer-Verlag.

[4] Kahng, A. B., Mandoiu, I., Pevzner, P., Reda, S., Zelikovsky, A. (2003): Engineering a scalable placement heuristic for DNA probe arrays. Proc. of the Seventh Annual Int. Conf. on Computational Molecular Biology, 148–83.

For more information, visit `http://gi.cebitec.uni-bielefeld.de/assb/chiplayout`

Presented at the 14th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB), Fortaleza, Brazil, August 6-10, 2006