Covering Points on a Circle with Circular Arcs

Vladimir Filkov*

June 10, 2002

1 Introduction

De-synchronization of cells in time-course expression experiments adversely effects the ability to analyze the data obtained. The question arises of how to retrieve cell cycle information from expression data of cell cycling genes in de-synchronized cells. Here we address this problem for the simplest case where gene expression is represented as a step function. For that case, we translate the problem into one of geometrical circle covering, and develop efficient algorithms to solve it.

If any individual gene's expression is approximated by a periodic step function, then the expression of that gene over all cells in the experiment is sum function of the individual step functions. When the cells are synchronized, as is the case in the beginning of any of the time-course gene expression experiments, this sum function is close to any of the individual functions. As the experiments progress, the cells get out of sync, and the resulting sum function no longer resembles a single step function. Instead it is a monotone rectilinear periodic signal. Our goal here is to de-convolve that periodic signal into its constituent step functions.

2 Problem Statement

It turns out that the problem of de-convolving a monotone rectilinear periodic wave signal into step (or square) functions is equivalent to finding a covering of points on a circle by arcs. Namely, if we sample a monotone rectilinear periodic signal at different times, and want to decompose the signal to basic step functions, we can wrap the sampling points around a circle, and try to cover them with periodic, basic, step functions. We may think of a periodic step function as having two states (over a single period): up or down. Each step function in its up-state can, therefore, be thought of as an arc covering points on a circle.

Thus, the problem of minimizing the number of step functions needed to reconstruct the original periodic rectilinear signal is equivalent to the problem of minimizing the number of arcs needed to cover points on a circle.

*Department of Computer Science, SUNY Stony Brook, Stony Brook, NY 11794-4400, vlfilkov@cs.sunysb.edu

Let $1, 2, \ldots, n$ be n points on the circle, where by "the circle" we will mean the unit circle. To the points we associate a set of heights, i.e. $\{h_1, h_2, \ldots, h_n\}$, indicating that point i is covered by h_i arcs. We define arcs in the conventional sense, i.e. an arc is a continuous part of the circle of any given angle, and for our purposes we treat the arcs as half-open line intervals on the circle (open at the beginning and closed at the end, clockwise). Thus, an arc covers a number of consecutive points which we call a *patch*. For our purpose of covering points on a circle, we consider two arcs to be equivalent if they cover the same patch on the circle. To a patch we associate a *patch vector* $\mathcal{V}$, where an entry $\mathcal{V}_i = 1$ if point i is in the patch, and 0 otherwise. Since each circular arc is continuous, the sequence of points it covers is consecutive, and the run of ones in the patch vector is circular, i.e. the ones occur consecutively, where the last and the first entries of the vector are considered consecutive.

In this paper we are considering the following problem:

Problem 1 *Cover n points on the circle with the minimal number of circular arcs, such that point i is covered by h_i arcs.*

In what follows, we consider the cases of:

- Arcs of fixed length (i.e. angle, $2\pi r$). We give a polynomial time algorithm for an arbitrary distribution of points on the circle, and a linear time algorithm for evenly distributed points, covered by same number of arcs of angle π.
- Arcs of variable length. We give an algorithm linear in the number of up/down transitions between points.

3 Arcs of Fixed Length

Problem 2 *Cover n points on a circle with the minimal number of fixed-length circular arcs, of angle $2\pi r, 0 < r < 1$, so that point i is covered by exactly h_i arcs, where $1 \leq i \leq n$.*

We identify arcs with patches since we are interested only in the points an arc covers.

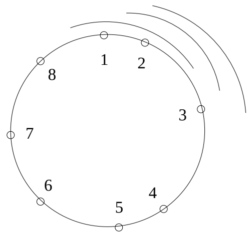

Figure 1: As an arc rotates around the circle, we get the patches $(1, 2), (2)$, and $(2, 3)$.

Problem 2 can be reformulated as an integer program once we realize that the total number of different patches around a circle of n points is linear in n. We show that:

Lemma 1 *There are $2n$ different patches on n points on a circle.*

Proof: Consider a unit, half-open, circular arc of radial angle $2\pi r$. It covers a sequence of points (i.e. a patch) on the circumference of the unit circle. As we rotate this arc clockwise around the center of the circle, the patches it covers change in a very particular way. Namely, if we think of a patch as a list, then as the arc rotates, a point is either added to the front, or removed from the back of the list, but not both at the same time, since the arc is half-open. As we rotate this arc, we enumerate the different patches it covers, starting from an arbitrary patch, which we call $\mathcal{P}_1$. We get patch $\mathcal{P}_{i+1}$ from patch $\mathcal{P}_i$ by either adding a point to $\mathcal{P}_i$ or removing a point from it. More formally, if $\mathcal{P}_i = (a_i, a_{i+1}, \ldots, a_{i+m-1})$ then $\mathcal{P}_{i+1}$ is either $(a_{i+1}, a_{i+2}, \ldots, a_{i+m-1})$ or $(a_i, a_{i+1}, \ldots, a_{i+m})$, where $1 \leq m, i \leq n$, and all indices are mod n.

An example of this discussion is given in Fig. 1.

The crucial observation here is that each of the n points on the circle will have been added exactly once and removed exactly once to/from the list of points covered as we rotate the arc. This is true, because if we add a point to the list for the second time, we will have made a full circle and repeat a patch. But this cannot be since all patches are distinct.

Every time we make a change to the list we get a new patch, and since we can make at most $2n$ changes to the list, we conclude that there are $2n$ different patches on n points on a circle. ▮

Now, we are ready to define the *covering matrix* A in the following way.

The *covering matrix*, A, is a 0-1 matrix of dimensions $n \times 2n$, with elements $a_{i,j} = 1$ if $p_i \in \mathcal{P}_j$ and 0 otherwise; where $1 \leq i \leq n$ and $1 \leq j \leq 2n$.

That is, the columns of the matrix are the patch vectors, where column j corresponds to the vector associated with $\mathcal{P}_j$, and the rows indicate if a point belongs to a patch or not.

A 0-1 matrix is called *column circular* (resp., *row circular*) if the 1's in its columns (resp., rows) occur consecutively, where the first and last entries are considered to be consecutive.

Lemma 2 *The covering matrix, A, is both column and row circular.*

Proof: The column circularity is clear since the patches cover the points continuously.

The row circularity follows from the ordering of the $2n$ patches. To see this assume the opposite, i.e. let a row R of A be non-circular. Then, R has at least two groups of zeros separated by ones, such that by rotation of the row, the 0's cannot be grouped in one group. But this means that point P_i, where i is the number of the row, $1 \leq i \leq n$, has been inserted at least twice to the list, as illustrated bellow, where the insertions are underlined.

Row i: $(1, \ldots, 1, 0, \ldots, 0, \underline{1}, 1, \ldots, 1, 0, \ldots, 0, \underline{1}, 1, \ldots, 1)$

But, this contradicts the fact that each point can only be inserted once to the list as we move from one patch to the next. ▮

Here is the matrix reformulation of the problem:

Problem 3 *Given a row-circular covering matrix A of dimensions $n \times 2n$, and a covering vector $S = \{s_1, s_2, \ldots, s_n\}$, solve the following integer program:*

$$\begin{array}{ll} \textit{minimize} & 1X \\ \textit{subject to} & \begin{array}{rcl} AX & = & S \\ X & \geq & 0 \textit{ and integer} \end{array} \end{array} \tag{1}$$

where $X = (x_1, x_2, \ldots, x_{2n})$ is a vector, such that x_i is an integer indicating how many times the patch $\mathcal{P}_i$ is to be used for the optimal covering, and $1X = x_1 + x_2 + \ldots + x_n$ is the total number of patches used in the covering (1 is the vector of all ones).

In [1], this problem is solved in polynomial time by using Network-Flow/LP algorithms. ▮

3.1 Evenly Distributed Points, Covered by Same Number of Arcs, Each of Length π

We say that n points on a circle are *evenly distributed* if the radial angle between any two consecutive points is the same, i.e. the angle is $2\pi/n$. In this section we consider the following simplification of the previous problem, which can be solved optimally.

Problem 4 *Cover n evenly distributed points on a circle with the minimal number of arcs k, each of radial angle π (half circles), such that point p_i is covered by exactly S arcs, for $1 \leq i \leq n$.*

In order to solve this problem, we prove the following theorem:

Theorem 1 *Let $(p_1, p_2, ..., p_n)$ be an ordered set of evenly distributed points on a circle. Then, k circular arcs of angle π cover the points so that each point is covered by exactly S arcs if and only if*
For n even, $2S/k = 1$; and
For n odd, $1 + 1/n \geq 2S/k \geq 1 - 1/n$.

Proof: We will distinguish two cases, based on the parity of n.

(1) Let n be even, i.e. $n = 2n'$.

Then, each circular arc covers exactly $n/2 = n'$ points.

To see this, we assume the opposite. Let A be an arc covering less(more) than n' points. Then, A spans an angle smaller(greater) than $(2\pi/n)n' = \pi$. Contradiction.

Then, it follows that all k arcs together cover $n'k$ points, and since each of the points is covered exactly S times for a total of Sn coverings, we have:

$$n'k = Sn \text{ or } k = 2S$$

An example of such a covering is the one where one half of the circle and all points on it are covered with S arcs, and similarly the other half of the circle.

(2) Suppose n is odd. To that end, let $n = 2n' + 1$.

Similarly as above, we prove that each circular arc, in this case, covers either $(n-1)/2 = n'$ or $(n+1)/2 = n' + 1$ points on the circle.

Assume the opposite.

- let A be an arc covering more than $n' + 1$ points. Then, A spans an angle larger than $(2\pi)/n \cdot (n'+1) = 2\pi/n \cdot (n+1)/2 = \pi(n+1)/n > \pi$. Contradiction.

- let A be an arc covering less than n' points. Then, A spans an angle smaller than $(2\pi)/n \cdot n' = 2\pi/n \cdot (n-1)/2 = \pi(n-1)/n < \pi$.

Following a similar argument as before, we see that all k arcs together cover the points at most $k(n'+1)$ times, and at least kn' times. Hence,

$$k(n'+1) \geq Sn \geq kn' \tag{2}$$

which works out to the following necessary condition:

$$1 + \frac{1}{n} \geq \frac{2S}{k} \geq 1 - \frac{1}{n} \tag{3}$$

Now, we need to prove that this condition is also sufficient.

Let k_1 be the number of arcs covering n' points, and k_2 be the number of arcs covering $n' + 1$ points.

Then, we have the following system:

$$\begin{cases} k_1, k_2 \geq 0 \\ k_1 + k_2 = k \\ k_1 n' + k_2(n'+1) = Sn \end{cases} \tag{4}$$

which solves to:

$$\begin{cases} k_1, k_2 \geq 0 \\ k_1 = k\frac{n+1}{2} - Sn \\ k_1 = Sn - k\frac{n-1}{2} \end{cases} \tag{5}$$

It is easy to see that the system (5) is equivalent to (3), because of

$$k_1 \geq 0 \Leftrightarrow k\frac{n+1}{2} - Sn \geq 0 \Leftrightarrow 1 + \frac{1}{n} \geq \frac{2S}{k}$$

and similarly for k_2.

Hence, if k_1 and k_2 are solutions of (5), by exhibiting a covering in which k_1 of the patches cover n' points, and k_2 of the patches cover $n' + 1$ points, we would have proved the sufficiency. To that end, we prove the following two easy lemmas.

Lemma 3 *For any m, where $0 < m \leq n$, n arcs covering m points on the circle each, can be arranged to cover each of the n points on the circle exactly m times.*

Proof: We will prove that arcs covering the following patches provide the needed covering:

$\mathcal{P}_1 = (p_1, p_2, \ldots, p_m)$, $\mathcal{P}_2 = (p_2, p_3, \ldots, p_{m+1}) \ldots$
$\mathcal{P}_n = (p_n, p_{n+1}, \ldots, p_{n+m})$
where all the indices are mod n.

Informally, patch $\mathcal{P}_i$ is rotated with respect to patch $\mathcal{P}_{i-1}$ clockwise, so that the leftmost point is eliminated and a point to the right is added.

The proof is by induction on m. For $m = 1$, the patches $(p_1), (p_2), \ldots, (p_n)$, consisting of one point each, obviously cover each point on the circle exactly once. Let the claim be true for $m = k$. That is, the patches

$\mathcal{P}_1 = (p_1, p_2, \ldots, p_k)$, $\mathcal{P}_2 = (p_2, p_3, \ldots, p_{k+1}) \ldots$
$\mathcal{P}_n = (p_n, p_{n+1}, \ldots, p_{n+k})$
cover the n points exactly k times.

Then the patches
$\mathcal{P}_1 = (p_1, p_2, \ldots, p_{k+1})$, $\mathcal{P}_2 = (p_2, p_3, \ldots, p_{k+2}) \ldots$
$\mathcal{P}_n = (p_n, p_{n+1}, \ldots, p_{n+k+1})$
cover the n points exactly $k+1$ times, since each patch covers one additional point than before (the last point in each), and all such points are different, making the total number of coverings per point be equal to $k+1$. ∎

Thus, n arcs, covering n' points (resp., $n'+1$ points) each, can be arranged to cover each of the n points on the circle exactly n' times (resp., $n'+1$ times).

Notice that $n | k_1 - k_2$, since from (5) we have $k_1 - k_2 = k\frac{n+1}{2} - Sn - (Sn - k\frac{n-1}{2}) = kn - 2Sn = n(k - 2S)$. Thus, we write $k_1 = q_1 n + r$ and $k_2 = q_2 n + r$, where $0 \leq r < n$.

Lemma 4 *$2r$ of the $k = k_1 + k_2$ patches above, can be arranged to cover each of the n points on the circle exactly r times, where $r = n$ mod $k_1 = n$ mod k_2.*

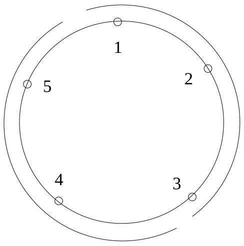

Figure 2: Two arcs in complementary position on a circle, one covering 2 and the other one 3 of the 5 points

Proof: By taking one pair of patches, one patch of which covers n' points on the circle, and the other one $n'+1$ points on the circle, and placing them in a complementary position on a circle, as shown in Fig. 2, we would have covered all points on the circle exactly once. So, by taking r such pairs (r patches of each kind), each point on the circle would be covered exactly r times. ∎

From the lemmas, the $k = k_1 + k_2$ patches can be arranged as follows to obtain the needed covering:

- $q_1 n$ of the patches that cover n' points each, can be arranged to cover all points $q_1 n'$ times;
- similarly $q_2 n$ of the patches that cover $n'+1$ points each can be arranged to cover all points $q_2(n'+1)$ times;
- $2r$ patches, r of which cover n' points each, and the other r cover $n'+1$ points each, can be arranged to cover all the points r times.

With this arrangement, each point on the circle is covered exactly S times, since:
$q_1 n' + q_2(n'+1) + r =$
$q_1 \frac{n-1}{2} + q_2 \frac{n+1}{2} + r =$
$\frac{1}{2}[n(q_1+q_2) - (q_1-q_2) + 2r] =$
$\frac{1}{2}[(k-2r) - (k-2S) + 2r] = \ldots = S$
Here we used the identities $n(q_1+q_2) = k - 2r$ and $q_1 - q_2 = k - 2S$ which can easily be obtained from system (5).

This proves Theorem 1. ∎

It can be easily proven now, that:

Corollary 1 *The minimal number, k, of arcs of size π needed to cover n evenly distributed points on the circle, such that each points is covered by exactly S arcs, is $2S$, when n is even, and $2S/(1+1/n)$ when n is odd.*

Proof: Follows directly from Theorem 1. ∎

Corollary 2 *Problem 4 is solvable optimally in time $\Theta(n+k)$.*

Proof: The input is $\Theta(n)$.

We output k arcs, following the covering construction described in the proof of Theorem 1. Since it obviously takes a constant number of steps per arc reported, the corrolary is proved. ∎

4 Arcs of Arbitrary Length

When the arcs' lengths are not fixed we show that the following problem can be solved optimally.

Problem 5 *Cover n points on a circle with the minimum number of circular arcs, such that point i is covered by h_i arcs.*

4.1 Definitions and Notation

Denote by C_{min} the minimal number of arcs needed to cover the circle. Let each point i be covered by h_i arcs. Let $h_{max} = MAX(h_1, h_2, ..., h_n)$.

If $h_{i+1} > h_i$, we say that there are $ut_i = h_{i+1} - h_i$ up transitions between points i and $i+1$, and if $h_{i+1} < h_i$ we say that there are $dt_i = h_i - h_{i+1}$ down transitions between points i and $i+1$. More succinctly, we write

$ut_i = h_{i+1} - h_i$, if $h_{i+1} > h_i$, and 0 otherwise, and

$dt_i = h_i - h_{i+1}$, if $h_{i+1} < h_i$ and 0 otherwise,

where all indices are between 1 and n (inclusive), and are taken mod n.

We let $T_{up} = ut_1 + ut_2 + ... + ut_n$, be the sum of all up transitions, and $T_{down} = dt_1 + dt_2 + ... + dt_n$ be the sum of all down transitions. It is obvious that $T_{up} = T_{down}$. So we write $T = T_{up} = T_{down}$.

Claim 1 *C_{min} is not less than $MAX(T, h_{max})$.*

Proof: Obviously C_{min} is not less than h_{max}. We need only consider, then, the case where T is greater than C_{min}. But this follows easily from the fact that each up transition (or down transition) indicates the beginning (ending) of an arc. Since we must have at least the number of arcs that have "begun" or started at some place on the circle, the claim follows. ∎

4.2 An Optimal Time Min-Circle-Cover Algorithm

Let $MAX(a,b)$ be the greater of two numbers a and b. Next, we give an algorithm which gives exactly $MAX(T, h_{max})$ number of arcs to cover the points on a circle. With that, our problem is solved optimally.

Algorithm 1 *Min-Circle-Cover*

Let each arc a_i have a start point s_i and an end point e_i, $i = 1, ..., n$.

1. *Cut the circle at h_{max}. We get a line segment with $n+1$ points on it, the first and the last point being identical and having identical heights of h_{max}.*
2. *Cover the points on the line segment with the following greedy scheme*
   (a) *Begin by using h_{max} arcs to cover the first point on the line.*
   (b) *In a left to right fashion, for all points i, $i = 2, ..., n+1$, do the following:*
      - *if $h_i > h_{i-1}$ start $h_i - h_{i-1}$(i.e. ut_{i-1}) new arcs at that point;*
      - *otherwise kill the "youngest" $h_{i-1} - h_i$ (i.e. dt_{i-1}) arcs.*
3. *Merge as many as possible arcs from the left end of the line with arcs from the right end, as follows:*
   - *Order the left h_{max} arcs in a list L (i.e. a_1, a_2, $\ldots, a_{h_{max}}$) by their ending time e_i in an ascending order (i.e. the youngest to have terminated first). These are the arcs covering the first point on the line.*
   - *Order the right h_{max} arcs in a list R by their starting time s_i in an ascending order (i.e. the oldest to have terminated first). These are the arcs covering the last point on the line.*
   - *Index the lists L and R by i and j respectively, where $i = j = 1$ in the beginning.*
     (a) *if $j > h_{max}$ done;*
     (b) *if L_i and R_j do not overlap, merge them, $i = i+1, j = j+1$;*
     (c) *else $j = j+1$;*

4.3 Analysis

Algorithm Min-Circle-Cover runs in $O(T + h_{max} + n)$ time. This is obvious, if we realize that the lists L and R can be filled in during step (2) of the algorithm.

Theorem 2 *Min-Circle-Cover yields $MAX(T, h_{max})$ arcs.*

Proof: Notice that in (2b) of Min-Circle-Cover, exactly T arcs have been started and exactly T arcs have been killed, since at every point i we start ut_i or kill dt_i arcs, thus a total of $h_{max}+T$ arcs exist in the beginning of step (3).

To finish the theorem, we show that in step (3) of the algorithm we merge $MAX(T, h_{max})$ arcs.

- Case 1: $T < h_{max}$

  Because the circle has been cut at h_{max}, by proceeding in a left to right fashion in step (2) of the algorithm, we must encounter a down transition before an up transition. That is, an arc has ended before a new one has started, and so it (the arc that has ended) can be merged with the longest non-overlapping one that survives to the right end of the line (this is the first arc from the L list). Also, since we started from h_{max}, the sum of the up transitions until a certain point will never be more than the sum of the down transitions until that point, and therefore for any of the T arcs that have started we can find an arc to merge, from the list of h_{max} arcs. Thus, from the total of $T + h_{max}$ arcs we can merge T of them and be left with h_{max}, which is $MAX(T, h_{max})$.

- Case 2: $T \geq h_{max}$.

  In a similar fashion, since the circle is cut at h_{max}, by proceeding from left to right, we come to a down transition before an up transition. That means that we can pair off the shortest arc from the left with the longest one from the right, because the one on the right started after the one on the left ended. Because we have more transitions than arcs that have initially started, we will be able to pair all h_{max} of them off (merge). Thus, after the merging, we are left with $T + h_{max} - h_{max} = T$ arcs, which is $MAX(T, h_{max})$ arcs.

∎

5 Conclusion and Future Work

In this paper we were interested in covering points on a circle, by the minimal number of circular arcs, so that each point is covered by a certain number of arcs. Our results are:

- Fixed-length arcs:
  - Points in a general position: We transformed the problem into an integer program, solved in polynomial time in the literature.
  - Evenly distributed points: We gave an optimal time $\Theta(n+k)$ algorithm (k is the minimal number of arcs).
- Arcs of arbitrary length: We gave an optimal, $O(n + T + h_{max})$ time algorithm.

An interesting question at the end is whether a more direct way of solving the fixed-length arcs/general distribution points problem, without resorting to the Network Flows/LP machinery. Also, it is an open problem to find the optimal time algorithm in that case.

As a future direction, we can extend our research on sets of arcs, where all of the arcs within a particular set are of fixed lengths, but lengths across sets can vary. Also, we can consider more interesting functions than step functions (like trigonometric functions) as the basic functions that comprise a given rectilinear signal.

References

[1] J. J. Bartholdi III, J. B. Orlin, H. D. Ratliff. "Cyclic Scheduling via Integer Programs with Circular Ones," *Operations Research* **28,** 1074–1085 (1980).

[2] R. K. Ahuja, T. L. Magnanti, J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, Upper Saddle River NJ, 1993.

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest. *Introduction to Algorithms.* MIT Press, Cambridge MA and McGraw-Hill NY, 1990