Gene expression

Optimal timepoint sampling in high-throughput gene expression experiments

Bruce A. Rosa¹,², Ji Zhang³, Ian T. Major², Wensheng Qin¹,* and Jin Chen²,4,*
¹Biorefining Research Institute and Department of Biology, Lakehead University, 955 Oliver Road, Thunder Bay, Canada ON P7B 5E1, ²MSU-DOE Plant Research Laboratory, Michigan State University, 612 Wilson Road, East Lansing, MI 48824, USA, ³Department of Mathematics and Computing, University of Southern Queensland (Toowoomba campus), West Street, Toowoomba QLD 4350, Australia and ⁴Department of Computer Sciences and Engineering, Michigan State University, 3115 Engineering Building, East Lansing, MI 48824, USA

Associate Editor: Trey Ideker

ABSTRACT

Motivation: Determining the best sampling rates (which maximize information yield and minimize cost) for time-series high-throughput gene expression experiments is a challenging optimization problem. Although existing approaches provide insight into the design of optimal sampling rates, our ability to utilize existing differential gene expression data to discover optimal timepoints is compelling.

Results: We present a new data-integrative model, Optimal Timepoint Selection (OTS), to address the sampling rate problem. Three experiments were run on two different datasets in order to test the performance of OTS, including iterative-online and a top-up sampling approaches. In all of the experiments, OTS outperformed the best existing timepoint selection approaches, suggesting that it can optimize the distribution of a limited number of timepoints, potentially leading to better biological insights about the resulting gene expression patterns.

Availability: OTS is available at www.msu.edu/~jinchen/OTS.
Contact: wqin@lakeheadu.ca; jinchen@msu.edu
Supplementary information: Supplementary data are available at Bioinformatics online.

Received on April 5, 2012; revised on June 11, 2012; accepted on August 13, 2012

1 INTRODUCTION

Time-series high-throughput gene expression experiments can measure the expression levels of tens of thousands of genes in a biological sample over time and provide dynamic information which can be used to construct regulatory networks and infer regulatory relationships among genes (Wang et al., 2008). Although there are several thousand time-series microarray and RNA-seq datasets on the Gene Expression Omnibus (GEO) database (Edgar et al., 2002), as of June 2012, most of these contain very few timepoints. Figure 1 shows that >75% of these datasets (in which ‘time’ has been set as a subset variable type) in GEO contain five or fewer timepoints. Given that researchers are often limited to being able to sample very few timepoints, it is extremely important to choose the most appropriate timepoints for observing strong target gene expression pattern changes. With a fixed number of samples, researchers can choose between (i) a very densely sampled short time-series experiment, in which important gene regulation events that do not occur quickly may be missed or (ii) a sparsely sampled long time-series experiment, where improperly positioned timepoints can lead to missing rapid but important regulation events and can also lead to temporal aggregation bias (which reduces the ability to infer actual regulatory relationships; Singh et al., 2005).

Determining the best sampling timepoints for sparsely sampled time-series high-throughput experiments is a challenging optimization problem that is frequently discussed in the biological literature (Chikina et al., 2009; Gustafsson and Hornequist, 2010; Marioni et al., 2008; Massonnet et al., 2010; Wang et al., 2008). An active learning algorithm has been developed for iteratively choosing timepoints to sample, using the uncertainty in the interpolation of the currently estimated time-dependent curve as the objective function (Singh et al., 2005). The performance evaluation in this study showed that this algorithm can find optimal timepoints such that majority cycling yeast genes can be identified.

However, to capture the differential gene expression patterns, the interpolation step requires a minimum of five timepoints (according to their online documentation), so it would not have been applicable for 75% of the existing datasets in GEO and would have only been able to predict very few timepoints in almost all of the existing datasets.

Furthermore, active learning is based only on the differential gene expression data in the dataset to which a new timepoint will be added, and existing time-series datasets using similar treatments (which may be high resolution and contain useful differential gene expression information) cannot be applied in the algorithm. Although other advanced gene expression prediction or interpolation methods can utilize sequence information (Beer and Tavazoie, 2004) and ‘biologically plausible’ constraints (Falin and Tyler, 2011) on gene expression estimates, these approaches do not address the complicated issue of timepoint selection among large groups of genes and also cannot utilize existing data.

In this article, we present a new model called Optimal Timepoint Selection (OTS) to identify optimal sampling timepoints for new microarray and RNA-seq experiments, based

*To whom correspondence should be addressed.
on gene expression data in existing datasets. We build OTS based on three observations: (1) Gene expression experiments can be sampled in an online fashion; i.e. samples can be treated and collected at a high rate and then stored at a relatively low cost, and particular samples can be measured at a later time, after deciding which timepoint will be optimal (Singh et al., 2005). (2) A researcher is usually interested in capturing the expression patterns of a subset of genes (which may be grouped into several clusters with similar expression patterns) associated with a given treatment/condition. and (3) Differential gene expression patterns from previous experiments performed under similar treatments/conditions can provide information valuable for defining an optimal timepoint for sampling, even if the sampling rates are different from the new experiment.

Based on these observations, a straightforward approach to choosing the best timepoint is to find unsampled timepoints at which there are significant upregulation or downregulation events for the genes of interest in the existing datasets. This approach is based on the assumption that the differential expression patterns for the genes of interest in existing datasets are similar to each other and are similar to the dataset to which a timepoint will be added. However, in practice, this assumption may be violated in many cases due to (1) large differences in the dynamic ranges between platforms (e.g. RNA-seq technology has a dynamic range several orders of magnitude higher than microarray technology; Marioni et al., 2008); (2) inconsistency among different datasets, either due to different growing conditions, different treatments or ‘lab signatures’, which result in differences in differential gene expression patterns among different laboratories, even after attempts to reproduce conditions exactly (Massonnet et al., 2010); (3) high noise rates in expression values, particularly for microarray datasets (Marioni et al., 2008) and (4) sparse sampling rates in existing data.

To address these data integration problems, we have developed OTS, which includes a novel method of combining differential gene expressions from existing datasets (‘training’ datasets) based on their similarity to the experiment to which timepoints will be added (‘current’ dataset). OTS is novel in the following ways:

(1) Projection of differential gene expression to threshold space: In contrast to existing differential gene expression prediction algorithms (Chikina et al., 2009; Falin and Tyler, 2011; Gustafsson and Hornquist, 2010), the goal of our method is to predict the best timepoints to add to a high-throughput experiment. Therefore, rather than focusing on specific expression patterns, we are instead interested in how many genes are significantly differentially expressed at each timepoint, and how significant the overall expression values are (in a categorized fashion). Consequently, we project the differential gene expression values to threshold space to better capture important regulatory timepoints (explained in Section 2.3).

(2) Data normalization and scaling: Instead of averaging or pooling all of the training data together, we first weight each training data’s contribution to the overall result based on their similarity to the current dataset. Then, we adjust the weighted-average values with a shifting function for local fitting (explained in Section 2.4).

(3) Timepoint selection with multi-objective optimization (MOO): We adopt a MOO model to select the overall optimal timepoint for all of the clusters (Coello, 1999). MOO is superior to the sampling voting method because timepoints chosen by MOO benefit all (or the majority) of clusters, while the sampling voting method may be biased to one or a few clusters (explained in Section 2.5).

The overall experimental approach for OTS is shown in Supplementary Figure S1 and Section S1. First, a biological experiment is performed, and samples are preserved at dense timepoints. A subset of timepoints (including at least the last timepoint in the range of interest and one other timepoint) is sampled. Then, time-series training datasets are collected. It is not necessary for the training datasets to be collected using the same technology (i.e. PCR, microarray or RNA-seq experiments), but they should use treatments or conditions that are expected to affect target treatment–response genes in the same way as in the current dataset. OTS produces a ranked list of the optimal timepoints to be selected next. The optimal timepoint(s) can then be sampled and added to the current dataset for the identification of the next optimal timepoint. This process can then be continued iteratively until all of the samples or all of the resources available for sampling are used up. This online-sampling approach is advantageous when studying organisms for which the sample collection step is significantly less expensive than the gene expression measurement step. For difficult or costly experiments (including clinical experiments), it is more logical to measure the gene expression in every available sample (Singh et al., 2005).

In the performance experiments in this study, OTS was applied using high-throughput time-series datasets for two different organisms (yeast and Arabidopsis) utilizing different platforms (microarray and RNA-seq). Noisy, sparsely sampled and poorly matched datasets were used as training. In all the experiments, OTS clearly outperforms the existing approaches.

2 METHODS

The goal of this article is to develop a computational algorithm to design the sampling rate of time-series gene expression experiments such that the real differential gene expression patterns for genes of interest are captured as accurately as possible.

Specifically, our approach is to generate an estimate dataset by integrating training data, and to identify the timepoint at which the estimate dataset is the most different from the current dataset, which may result in the identification of the most significant differential regulation events missing in the current dataset.
Mathematically, given training datasets \( R = \{R_1, R_2, \ldots, R_m\} \), a current dataset \( U \) (with differential gene expression values available at timepoint set \( T_U \) and unmeasured biological samples available at timepoint set \( T_{\text{opt}} \)), rank the optimal timepoints in the timepoint set \( T_{\text{opt}} \), such that the best-ranked timepoints minimize the difference between the interpolated and real differential gene expression curves for all genes of interest \( G \). The outline of OTS is shown in Algorithm 1.

Algorithm 1 Optimal Timepoint Selection

**Input:**
- \( U \): current differential gene expression dataset
- \( R \): set of training differential gene expression datasets
- \( G \): set of genes of interest
- \( H \): threshold number
- \( T_U \): set of timepoints measured in \( U \)
- \( T_{\text{opt}} \): set of available but unmeasured timepoints

**Output:** \( T_{\text{opt}} \): set of ranked optimal timepoints

1. \( C \leftarrow \text{clustering}(G, R) \)
2. \( Q \leftarrow \emptyset \)
3. for all cluster \( c \in C \) do
   4. \( I_R \leftarrow \text{DataInterpolation}(R, T_U) \)
   5. \( I_U \leftarrow \text{DataInterpolation}(U, T_U) \)
   6. \( D_R \leftarrow \emptyset; D_U \leftarrow \emptyset \)
   7. for all \( i \in T_U \cup T_{\text{opt}} \) do
      8. \( D_R \leftarrow D_R \cup \text{Thresholding}(I_R, H, i) \)
      9. \( D_U \leftarrow D_U \cup \text{Thresholding}(I_U, H, i) \)
   10. end for
   11. \( Q \leftarrow Q \cup \text{NormalizationAndScaling}(D_R, D_U) \)
12. end for
13. \( T_{\text{opt}} \leftarrow \text{MultiObjectiveOptimization}(Q) \)
14. return \( T_{\text{opt}} \)

2.1 Case study

For demonstration purposes, we used an *Arabidopsis* coronatine-treatment dataset as a case study (Fig. 2A). This dataset was produced to determine the effect of the phytotoxin coronatine (a molecular mimic of the plant hormone jasmonate) on global gene expression (manuscript in preparation). We used this densely sampled (21 timepoint) RNA-seq dataset as a mock ‘current’ biological experiment (first row in Fig. 2A), and a number of existing microarray datasets involving coronatine/jasmonate treatment as training datasets (see Section 3.1; Chung et al., 2008; Wierstra and Kloppstech, 2000). This case study starts with six timepoints, at 0.25 h (the first), 24 h (the last), and 1, 2, 3 and 5 h (selected iteratively by the first four rounds of OTS selection), and the fifth round of timepoint selection will be outlined in detail here (see Section 3.3).

2.2 Differential gene expression clustering and interpolation

As the first step of OTS (line 1 in Algorithm 1), all of the genes of interest \( G \) are clustered using K-means clustering [Demml and Kastner, 2003; implemented in Cluster 3.0 (http://rana.lbl.gov/EisenSoftware.htm); Eisen et al., 1998] based on their differential gene expression values in the training datasets \( R \), which produces \( C \), the set of clusters. In the case study, the coronatine-responsive genes were separated into 10 clusters; the fifth round of timepoint selection will be outlined in detail here (see Section 3.3).

To estimate the differential gene expression patterns at all timepoints, OTS linearly interpolates the values from the current \( U \) and the training \( R \) datasets to every available timepoint in \( T_U \) (Algorithm 1, lines 4 and 5; Meijering, 2002). Linear interpolation was used in order to minimize the inference of false peaks and valleys in the expression data (Benesty et al., 2004) and to avoid over-smoothing unevenly spaced timepoints, which occurs on sparsely sampled datasets when using other common interpolation methods (Meijering, 2002).

2.3 Projection of differential gene expression to threshold space

In the second step of OTS, rather than focusing on specific expression patterns, we capture important regulatory timepoints by measuring how many genes are significantly differentially expressed at each timepoint, and how significant the overall expression values are (in a categorized...
least-squares estimation), the challenge is that the difference between diff-
licult problem because the training DRC datasets may not be similar.

The largest difference is the optimal timepoint for each cluster. However,
the training datasets from the CTE table, and the difference between the
average and the standard deviation values, respectively, for the differen-
tation values which are higher (or lower) than a series of differential regu-
lation values; Chikina et al., 2009; Falin and Tyler, 2011; Gustafsson and
Hornquist, 2010), we project the differential gene expression data of
each cluster into threshold space, where the values for a given timepoint
are determined based on how many genes have differential gene expres-
sion values which are higher (or lower) than a series of differential regu-
lation thresholds (Fig. 3B; Algorithm 1, lines 6–10). This thresholding
process reduces noise in the comparison among datasets by ignoring
small fluctuations in differential gene expression value patterns while
capturing the overall pattern of the larger gene expression changes at
various magnitudes.

To avoid the bias introduced by setting only one regulation threshold
value, multiple evenly spaced positive and negative differential regu-
lation threshold values are defined to determine the degree to which a cluster
of genes is differentially regulated at a given timepoint \( t \), according to
Equation (1). Given a user-defined threshold number \( H \), we divide the
threshold space (3 SDs above and below the average differential gene
expression value) into two \( H \) sections. For example in Figure 3B, an
\( H \)-value of 6 has been used, and threshold values are shown.

Mathematically, to perform thresholding for a gene \( g \) with an expres-
sion value at timepoint \( i \) in dataset \( R_t \), we compute its differential regu-
lation count (DRC) by counting how many thresholds it is higher (or
lower) than if it is up- (or down-) regulated. This represents the DRC
for timepoint \( i \) in dataset \( R_t \) \( (D^i; \text{Equation } (1)) \). Higher DRC numbers
indicate stronger differential regulation, regardless of whether the genes
are upregulated or downregulated. The use of multiple thresholds ensures
that changing patterns in experiments with different dynamic ranges are
captured. For example, in Figure 3C, one gene crosses the top upregula-
tion threshold (at 3.29). These counts are made for each regulation threshold
and summed (Equation (1)). DRC curves for all the training and current
data are upregulated or downregulated. The use of multiple thresholds ensures
that the final estimate DRC dataset values are equal to the current data-
set’s contribution to the overall result based on their similarity to the
current DRC dataset in each cluster using NNLS regression (Chen et al.,
2010; Lawson and Hanson, 1995). Mathematically, given a \( n \times m \) matrix of DRC values derived from the training DRC datasets
\( D^i \), and an \( n \times 1 \) vector of DRC values derived from the current
DRC dataset \( \hat{D} \), a non-negative \( m \times 1 \) weight vector \( w \) is calculated,
which minimizes the difference between weighted training and current
DRC datasets (i.e. \( w = \arg \min ||D - \hat{D}||^2 \)). This weight vector \( w \) is
then used to calculate a weighted-sum NNLS estimate DRC dataset
(Fig. 3E). By forcing all of the weight values to be non-negative, it
avoids a problem introduced by standard LSE regression, wherein negative
weights can ‘flip’ the patterns, changing peaks to valleys and providing
false information in the estimation. This step also results in normalization
of experiments with different dynamic ranges.

In the second step (local fitting), in order to correct the NNLS estimate
fit, NNLS-weighted sum DRC values are shifted for each timepoint, such
that the final estimate DRC dataset values are equal to the current data-
set DRC values at every sampled timepoint \( T_i \) (indicated by vertical
dashed grey lines in Fig. 3F). The rest of the timepoints in the
NNLS-weighted estimate DRC dataset are shifted by an amount sug-
gested by the sampled timepoints and modulated by their distance from
the sampled timepoints according to a sigmoid weight distributed (Chen and
Mangasarian, 1995; Marler et al., 2006).

In summary, the estimate value at timepoint \( t_i \) \( (\hat{D}^i) \) is defined as

\[
\hat{D}^i = \begin{cases} 
\hat{D}^i + \frac{1}{m} \sum_{j=1}^{m} w_j (\hat{D}^i - D^j) & \text{if } t_i \in T_S \\
\sum_{j=1}^{m} w_j D^j & \text{otherwise}
\end{cases}
\]

where

\[
t_i = \arg \max_{1 \leq t_j \leq \hat{T}_S} |\hat{D}^i - \sum_{j=1}^{m} w_j D^j| \quad \text{and } t_j \text{ is a timepoint in the interpolated current dataset } (T_j \cup T_k), \hat{D}_i \text{ is the DRC value for timepoint } t_i \text{ in the current DRC dataset } (\hat{D}), D^j \text{ is the DRC value for timepoint } t_j \text{ in training DRC dataset } (R_j) \text{ and } w_j \text{ is the weight assigned by NNLS for training data } R_j. \text{ In the fraction component of this equation, the numerator calculates the largest observed shift (i.e. the largest amount of disagreement between the NNLS estimate and the sampled timepoints in the current dataset), which occurs at timepoint } t_i. \text{ The denominator then reduces the amount of this shift for the given timepoint } t_i, \text{ such that the shift will be smaller if there is more distance between } t_i \text{ and } t_j.

<table>
<thead>
<tr>
<th>Timepoint</th>
<th>Training datasets</th>
<th>Current dataset</th>
<th>Estimate DRC dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_1 )</td>
<td>( D_{11} ) ( \ldots )</td>
<td>( \hat{D}_1 ) ( \hat{D}_1 )</td>
<td>( \hat{D}_1 ) ( \hat{D}_1 )</td>
</tr>
<tr>
<td>( t_2 )</td>
<td>( D_{21} ) ( \ldots )</td>
<td>( \hat{D}_2 ) ( \hat{D}_2 )</td>
<td>( \hat{D}_2 ) ( \hat{D}_2 )</td>
</tr>
<tr>
<td>( \ldots )</td>
<td>( \ldots ) ( \ldots )</td>
<td>( \ldots ) ( \ldots )</td>
<td>( \ldots ) ( \ldots )</td>
</tr>
<tr>
<td>( t_n )</td>
<td>( D_{n1} ) ( \ldots )</td>
<td>( \hat{D}_n ) ( \hat{D}_n )</td>
<td>( \hat{D}_n ) ( \hat{D}_n )</td>
</tr>
</tbody>
</table>

Each row represents a timepoint available for sampling \( (T_1 \ldots T_n) \), and there are columns for the training \( (R_1 \ldots R_m) \), current \( (\hat{D}) \) and estimate \( (\hat{D}) \) datasets.
2.5 Timepoint selection with MOO

By clustering all of the genes based on their expression patterns and comparing the estimate and current DRC datasets, we are able to rank all of the timepoints for one cluster using curve difference scores ($Q^i$). However, if the ranks for each timepoint are different in different clusters, a cross-cluster ranking method is needed to rank timepoints for the entire dataset. Instead of applying a sampling voting method (used in Singh et al., 2005) which may be biased towards optimal timepoints in one or few clusters, OTS applies a MOO model to rank optimal timepoints which will most benefit all of (or the majority of) the clusters (Algorithm 1, lines 13 and 14; Coello, 1999).

Mathematically, MOO computes a $\lambda$-score (indicating optimality) for each timepoint. First, $\lambda$-dominance is determined for each timepoint pair as follows: we say timepoint $t_1$ $\lambda$-dominates timepoint $t_2$ (denoted as $t_1 \succ t_2$) if $Q^i$ is larger than $Q^j$ in $\lambda$ clusters, where $1 \leq \lambda \leq |C|$. For example, in Figure 4A, the $Q$-values for every cluster in the 12-h column are larger than the $Q$-values for all of the 10 clusters in the 6-h column, so the 12-h timepoint $\lambda$-dominates the 6-h timepoint at $\lambda = 10$. Second, the $\lambda$-score of a timepoint $i$ is defined as the number of other timepoints that $i$ $\lambda$-dominates, according to:

$$\lambda - \text{score}(i, \lambda) = \left\{ i \mid i \neq i, i \in T_A, i > i \right\},$$

where $i$ is a timepoint in $T_A$ and $i$ is any other timepoint in $T_A$.

Optimal timepoints are selected by ranking based on the $\lambda$-score values of the timepoints. Initially, $\lambda$ is set to the number of clusters ($|C|$), but if two or more timepoints share the same $\lambda$-score (such as 1.5, 8, 10 and 14 h in the first row of Fig. 4B) then they are compared at $\lambda = |C| - 1$ (where, in the second row of Fig. 4B, timepoint 1.5 outranks the others to get a second-place overall rank). If there remains a tie, then they are compared at $\lambda = |C| - 2$, and the process is repeated until each timepoint is ranked. Using the final ranked timepoint list, researchers are free to sample one or more of the top-ranked timepoints in their biological experiment.

3 EXPERIMENTAL RESULTS

Three main experiments were used to evaluate the performance of OTS, and the performance was compared with uniform distribution and active learning timepoint selection (where applicable) (Singh et al., 2005). In the first experiment (which uses the Arabidopsis datasets described in Section 3.1), only the first and last timepoints from the current dataset were used as initial input, with five additional optimal timepoints added one-at-a-time, to simulate ‘iterative-online sampling’ on an initially very sparse dataset. This first experiment was re-run three times with different parameters to demonstrate the effectiveness of OTS when using lower quality training datasets and different gene selection methods. A second ‘iterative-online’ sampling experiment was run with the yeast datasets (also described in Section 3.1). For the third experiment (which also used the yeast datasets), we start with five evenly distributed timepoints (at 5, 30, 60, 90 and 120 min), and then add two more timepoints as a batch to ‘top-up’ the timepoints sampled, simulating the situation of choosing extra timepoints after conducting initial sampling determined by researcher’s knowledge/intuition.

As a comparison, Singh et al.’s active learning algorithm was also used to choose optimal timepoints, using the same number of clusters as OTS. Active learning requires at least five timepoints as initial input, so it was used for the top-up experiment, but for the iterative-online experiments (which start with only two timepoints) the first three selected timepoints were chosen using a uniform distribution across the time series. Random timepoint selection was also performed, where timepoints were randomly selected within the time range of each experiment 250 times.

3.1 Datasets

OTS performance was tested on differential gene expression (fold change) datasets from two different organisms. The first was Arabidopsis, for which certain gene functions are well studied but dense time-series differential gene expression datasets are difficult to find. The current dataset used was from a high-resolution (20 timepoints) coronatine-treatment RNA-seq experiment. Coronatine is a toxin produced by Pseudomonas syringae pv. tomato DC3000 and is a molecular mimic of the jasmonate hormone which mediates wound response in Arabidopsis (Thilmony et al., 2006). So, several different training datasets utilizing coronatine (three timepoints), Pst. DC3000 (two timepoints; Thilmony et al., 2006 and three timepoints; Kilian et al., 2007) and wounding treatments (six timepoints; Kilian et al., 2007) were used in this study (Fig. 2A). For this experiment, known jasmonate-responsive genes and circadian clock genes were selected as target genes (since the circadian clock influences jasmonic acid pathway activation; Goodspeed et al., 2012), based on the GO-SLIM categories ‘response to jasmonic acid synthesis’ (GO:0009753, 139 genes) and ‘circadian rhythm’ (GO: 0007623, 76 genes), for a total of 195 genes common to all the datasets.

The second organism tested was yeast. Here, a 25-timepoint microarray dataset which used α-factor treatment to synchronize cell cycles to the G1 phase was used as the current dataset (Pramila et al., 2006). Three other α-factor treatment datasets (25 and 12 timepoints; Pramila et al., 2006, and 17 timepoints; Spellman et al., 1998), and one dataset in which temperature changes also synchronized the cells to the G1 phase (12 timepoints; Cho et al., 1998) were used as training datasets.
(Fig. 2B). All of the genes common to the datasets in the GO category ‘mitosis’ (i.e. cell division; GO:0007067, 90 genes total; Ashburner et al., 2000) were used.

In the Arabidopsis experiment, very large differential gene expression values were expected for jasmonic acid response, based on the literature (Chung et al., 2008; Wierstra and Klopstech, 2000), and very low levels of noise are expected in the RNA-seq dataset (Marioni et al., 2008), so a threshold number \( H \) of 6 was used to preferentially capture these larger changes in expression. In contrast, for yeast experiments, a threshold number \( H \) of 3 was used in order to reduce the high expected noise in the datasets (Cooper and Shedden, 2003), by ignoring the small fluctuations in differential gene expression measurements. The Arabidopsis and yeast datasets were grouped into 10 and 8 clusters, respectively.

For these organisms, suitable training datasets were readily available; However, when there is not enough training data for an organism of interest, datasets from closely related organisms may be used, and homologous genes can be found in the target organisms using sequence similarities (She et al., 2009). Another potential approach for preparing training datasets is to utilize time-alignment algorithms on the datasets obtained from similar experiments. For example, cell-cycle patterns can be synchronized by shifting and stretching the time axis to align the time-series expression patterns of key cycling genes between datasets (Aach and Church, 2001). On the other hand, if there are many training datasets, a pre-screening approach for selecting appropriate training datasets is required. All these approaches could be utilized as pre-processing steps, but the first two have not been tested in this study in order to minimize the complexity of our experimental approach, and the third approach is introduced in Supplementary Section S3 and Figure S5.

### 3.2 Performance measurement

For performance evaluation, given the differential gene expression (fold change) data of a gene \( g \) in the current dataset, its predicted differential gene expression values at every unsampled timepoint were determined using linear interpolation. A measure of error between these interpolated and the actual differential gene expression values for all of the genes in \( G \) was derived, such that larger errors result from measurements with (1) poor agreement between the actual and predicted values and (2) large actual differential expression. These errors are summed for all genes and all timepoints and compared with the summed error at the start of the experiment to calculate the ‘percentage sum error’ \( (E_r) \) in the experiment:

\[
E_r = \frac{\sum_{g \in G} \sum_{i=1}^{T_d} |\hat{e}_{gi} - (\hat{e}_{gi} - e_{gi})|}{\sum_{g \in G} \sum_{i=1}^{T_d} |\hat{e}_{gi} - e_{gi}^0|}, \tag{4}
\]

where \( r \) is the round of timepoint sampling, \( e_{gi} \) is the actual differential gene expression value, \( \hat{e}_{gi} \) is the predicted differential gene expression value for a given round of sampling \( r \) and the predicted differential gene expression value at the start of the experiment, respectively, for gene \( g \) in the set of genes \( G \) at timepoint \( i \) \((1 \leq i \leq |T_d|)\). This equation is biased towards large errors for false negatives as opposed to false positives; That is, if we predict a small fold change for a gene which is actually strongly differentially regulated (false negative), then we miss an important biological event and the timepoint selection was poor. However, in the opposite case, if we predict a large fold change for a gene which is not actually differentially regulated (false positive) and we choose to sample that timepoint, we may waste a sample, but have not missed an important regulation event for that gene.

Error plots after the addition of the first timepoint in the iterative-online experiments and after the addition of both of the timepoints in the top-up experiment are shown in Figure S5A–C.

### 3.3 Performance analysis

In the Arabidopsis iterative-online experiment (Fig. 6A), the addition of the first OTS timepoint (at 5h) reduces the percentage sum error by 50%, which is much better than the timepoint selected by uniform distribution (12h, 32%). Figure 5A shows that after the addition of this first timepoint, compared with OTS, uniform distribution selection predicts that many genes are unchanged or downregulated at timepoints where they are actually strongly upregulated (circles near the bottom right of the plot, in
dark-shaded areas). The second timepoint added by OTS (at 2 h) further reduced the error by 35% of the initial error, compared with uniform distribution, which still has 66% of the initial error and still misses many strong upregulation events (Supplementary Fig. S2C). The early bias in the timepoint selection by OTS (which sampled at 1, 2, 3, 5 and 12 h) more effectively captures the early peaks expression levels of coronatine-induced genes (Chung et al., 2008) than even timepoint distribution (6, 12 and 18 h) followed by active learning (0.5 and 20 h; Supplementary Fig. S3A). This experiment demonstrates the effectiveness of OTS across platforms, and when using sparsely sampled training datasets with slightly different biological treatments.

In the iterative-online yeast experiment, OTS outperformed uniformly distributed timepoint selection (Fig. 6B), reducing the error by 35% after the addition of just one timepoint at 20 min (compared with only 5% reduction at 60 min in uniform distribution). The error plot in Figure 5B shows that the strongest differential regulation events are much more accurately defined by the timepoint selection in OTS (as shown by an abundance of even-timepoint spacing/active-learning circle marks in high-error areas in the bottom left and bottom right of the plot). After the addition of two timepoints selected by OTS (at 20 and 50 min), the initial percentage sum error is reduced by 47%, compared with just 14% reduction from uniform distribution-selected timepoints (Fig. 6B and Supplementary Fig. S2F). At the end of the experiment, the initial error is reduced by 56% using OTS timepoints (20, 35, 40, 50 and 95 min) compared with 51% using uniform distribution (30, 60, and 90 min) and active learning (25 and 95 min) timepoints (Supplementary Fig. S3B). Like in the Arabidopsis experiment, OTS outperforms active learning at every number of timepoints tested, and there is also a bias towards early timepoints, probably due to stronger and more co-ordinated cyclic gene responses immediately after synchronization (Cho et al., 1998; Spellman et al., 1998), and because many yeast cell cycle genes peak in late G1 (at ~20 min), the point at which the cell needs to ‘decide’ whether to divide or to continue to grow (Rodriguez-Sanchez et al., 2011). Additional potential biological insights provided by using OTS in both the Arabidopsis and yeast experiments are outlined in Supplementary Figure S4 and Section S2 (Chen et al., 1999; Fernandez-Calvo et al., 2011; Martinez et al., 2006; Xie et al., 1998).

OTS also selects early timepoints in the top-up yeast experiment (10 and 20 min), reducing the error by 26%. In this experiment, active learning adds timepoints at 25 and 95 min, and only reduces the error by 14%. The error plot in Figure 5C shows that the strongest upregulation events (on the right of the plot) are more accurately defined by the early timepoint selection in OTS, resulting in much stronger performance. Error plots for the start of each of the experiments and after the second round of the iterative-online experiments are shown in Supplementary Figure S2. These yeast experiments demonstrate strong performance for OTS despite a great deal of noise, because cell cycles are only weakly reproducible (even between replicates), α-factor synchronization and temperature treatments may elicit stress-related gene responses (Cooper and Shedden, 2003), and a very diverse set of genes is responsible for mitosis, which was the gene group selected here (Cho et al., 1998).

The robustness of OTS was tested using the Arabidopsis experiment setup (1) against different training sets and (2) against different predefined genes of interest. First, the coramine treatment microarray training dataset (in which the laboratory conditions were exactly the same as in the current experiment, making it the most closely matched dataset) was removed, to test whether OTS performance would be significantly affected. The results in Figure 6C show that the removal of this best-matched dataset only slightly reduced performance (2.6% average decrease for all of the timepoints selected), showing that even with only the other three training datasets, which use similar but not identical biological treatments (i.e. wounding and DC3000 treatments), OTS is still effective.

Next, a six-timepoint cold-treatment dataset (Kilian et al., 2007) was added to the four training datasets, to test whether adding poorly matched training data would negatively affect OTS performance. Cold treatment is appropriate for this test, as it functions through the DREB1/CFB transcriptional stress–response module, which is biologically and experimentally unrelated to the JAZ-MYC/Myb transcriptional modules activated by the coronatine/wounding response (Shinozaki et al., 2007). Again here, there was only a very slight reduction in performance with the inclusion of this poorly matched dataset (2.9% average decrease for all the timepoints selected), demonstrating the robustness of OTS against using poor training datasets, due to the assignment of relatively low weight values by the NNLS weighting step. A third training dataset test experiment was run, in which the well-matched coronatine treatment microarray experiment was removed and the poorly matched cold experiment was
added to the training datasets. Here, there was only a 2.0% decrease in performance for the first timepoint, but the later timepoints had a slightly >5% decrease in performance (Fig. 6C).

The results of an unsupervised gene selection method are shown in Figure 6D. Rather than using the knowledge-based gene ontology gene selection approach used in the other experiments, the top 100 genes with the highest fold change values across all four training datasets were selected. Here, three genes were removed from the analysis due to zero-control values in the current RNA-seq dataset (which result in undefined fold change values), and 8 clusters were used instead of 10 due to the smaller gene set size. This 97-gene set had little overlap (12 genes) with the knowledge-based gene selection set. Figure 6D shows that the performance of the two different gene selection strategies is similar, suggesting that an unsupervised gene selection approach could be used to fully automate OTS when knowledge-based gene categorizations are not applicable.

In summary, these tests indicate that OTS is able to learn sampling rates from suboptimal training data, its performance is robust against using irrelevant training data and it is compatible with automatic gene selection methods.

4 CONCLUSION

We have demonstrated that OTS can out-perform existing algorithms at finding optimal timepoints for defining true differential gene expression patterns for large groups of target genes. We have shown that this algorithm is robust to the use of sparsely sampled, poorly matched and cross-platform data, as well as to noise in the datasets. Because it utilizes existing data effectively, OTS can be applied on datasets starting with as few as two timepoints, in contrast to the active learning algorithm which requires a minimum of five timepoints as input (Singh et al., 2005).

As high-throughput gene expression measurement technologies continue to be developed, high-resolution sampling may eventually become cost-effective. For example, ‘nanostrings’ are a recently developed medium-throughput gene expression measurement technology capable of measuring up to 800 genes at once at a relatively low cost (Brumbaugh et al., 2012). However, using this technology, not all of the genes in the organism can currently be sampled, and the gene list needs to be pre-defined. Since OTS simply uses gene differential expression values as input, it would be possible and very advantageous to use the results from nanostring or real-time PCR experiments as training data for OTS, to select optimal timepoints. For RNA-seq technology, highly multiplex sampling is becoming increasingly accurate, allowing for denser timepoint sampling with a moderate increase in cost (Islam et al., 2011). As more time-series datasets are produced due to these advances in technology, more and better training datasets for OTS will be produced, and the demand for better knowledge-based timepoint selection methods will increase.

In this article, OTS was tested only using differential gene expression values, but it could also be extended to use other types of data, including raw transcript number counts, relative protein quantities or any type of measurement that can be sampled in an online fashion. Overall, OTS can be used to significantly improve the results from biological experiments by allowing researchers to optimize the distribution of timepoints when there is a limit on the number of samples that can be measured across a time-series dataset.

The estimation power of extrapolation of time-series gene expression data is much less than for interpolation, particularly for relatively simple linear extrapolation methods (Haye et al., 2012). For this reason, OTS is currently limited to selecting timepoints within the time range available in training datasets. Eventually, more sophisticated extrapolation methods such as the non-linear differential equation models outlined in Haye et al. (2012) may be integrated to improve the predictive power of OTS.

Funding: This project has been funded by the U.S. Department of Energy (Chemical Sciences, Geosciences and Biosciences Division, grant no. DE-FG02-91ER20021 to J.C. and Natural Sciences and Engineering Research Council of Canada Research Development Fund, Canada to W.Q.

Conflict of Interest: none declared.

REFERENCES


