



US 20040014044A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2004/0014044 A1**

Scott et al. (43) **Pub. Date: Jan. 22, 2004**

(54) **METHOD FOR DETERMINING
BIOLOGICAL EXPRESSION LEVELS BY
LINEAR PROGRAMMING**

(52) **U.S. Cl.** **435/6; 702/20**

(75) **Inventors: Ridgway Scott, Chicago, IL (US);
Stephen J. Wright, Madison, WI (US);
Stuart A. Kurtz, Homewood, IL (US);
Terry Clark, Chicago, IL (US); Chris
(Hristem) Dyanov, Chicago, IL (US);
Richard Quigg, Western Springs, IL
(US)**

(57) **ABSTRACT**

A method for determining a matrix of expression levels corresponding to a set of biological targets (e.g., genes or gene fragments) and a set of biological samples, including obtaining a matrix of signal values corresponding to the set of biological targets; computing a vector of expression levels for a sample in the set of biological samples using the matrix of signal values; storing the vector of computed expression levels in a storage matrix; repeating the computing and storing steps for each sample in the set of biological samples; and outputting the storage matrix as the matrix of expression levels. The method, based on a linear programming formulation of the problem, works for both "promiscuous" probe array data, in which there may be multiple targets indicated by a single probe, and the "polygamous" case, in which there are multiple probes for a single target. The preferred method can also process data obtained from multiple SAGE analyses using multiple markers. A second embodiment of the method determines optimal expression levels when the available probe data is noisy or uncertain.

Correspondence Address:

**OBLON, SPIVAK, MCCLELLAND, MAIER &
NEUSTADT, P.C.
1940 DUKE STREET
ALEXANDRIA, VA 22314 (US)**

(73) **Assignee: University of Chicago, Chicago, IL (US)**

(21) **Appl. No.: 10/198,141**

(22) **Filed: Jul. 19, 2002**

Publication Classification

(51) **Int. Cl.⁷ C12Q 1/68; G06F 19/00;
G01N 33/48; G01N 33/50**

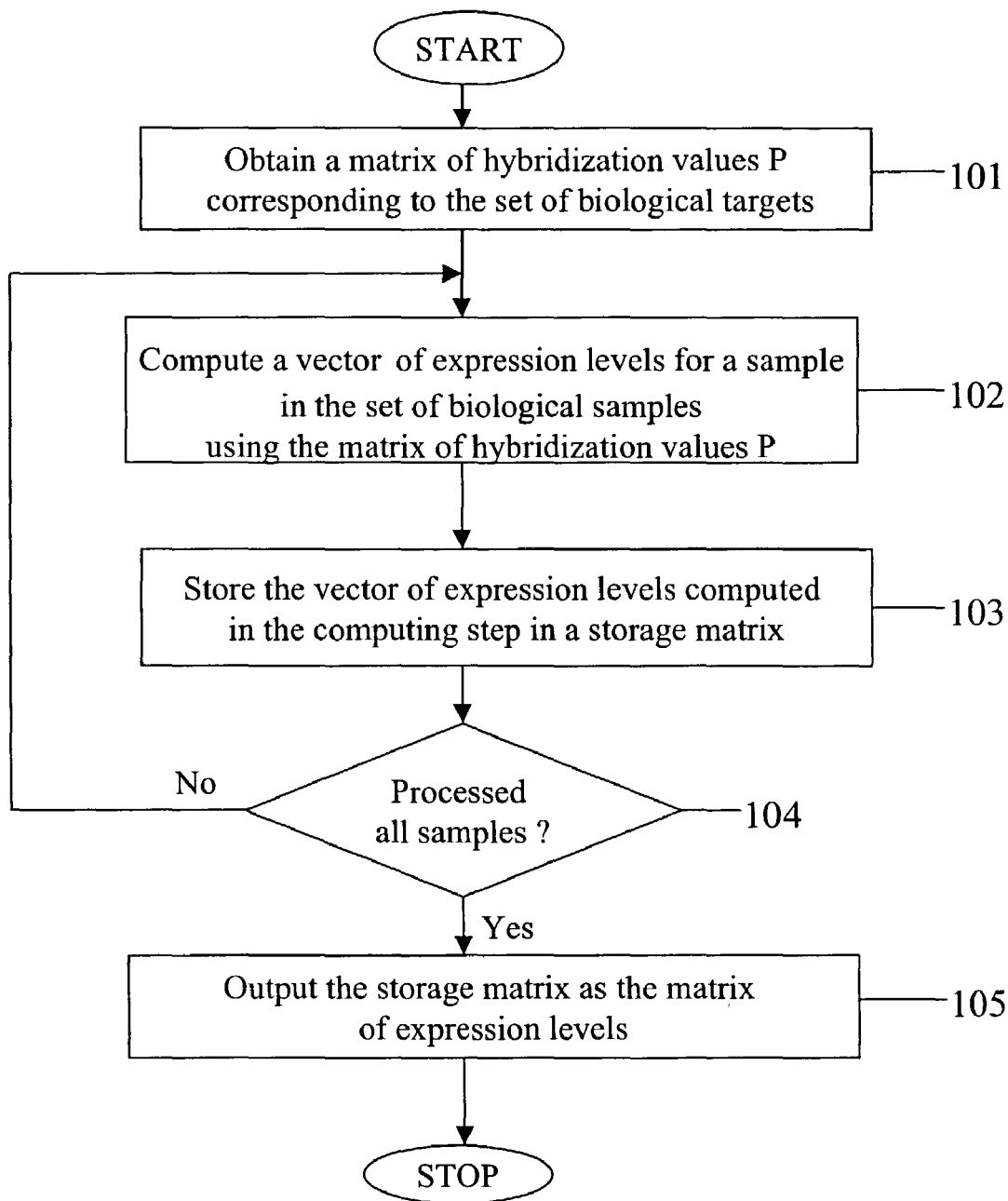


FIG. 1

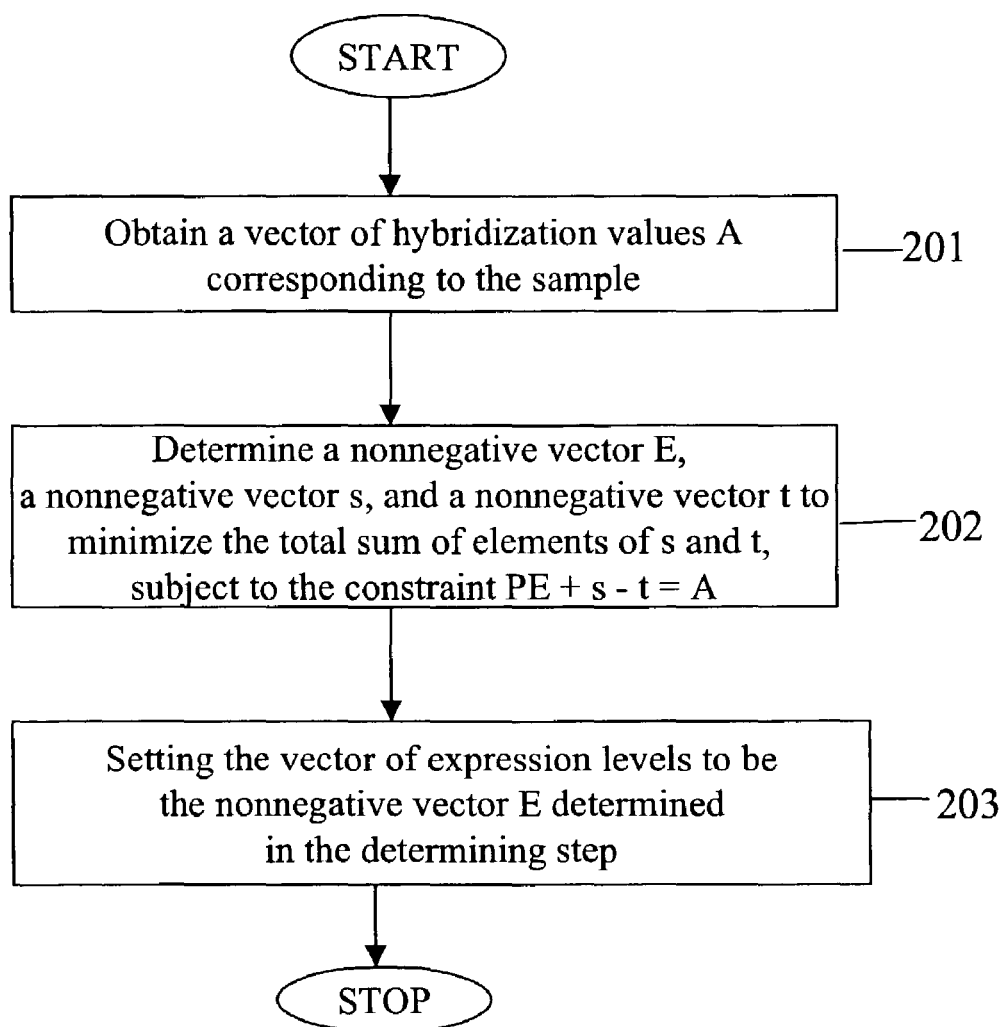


FIG. 2

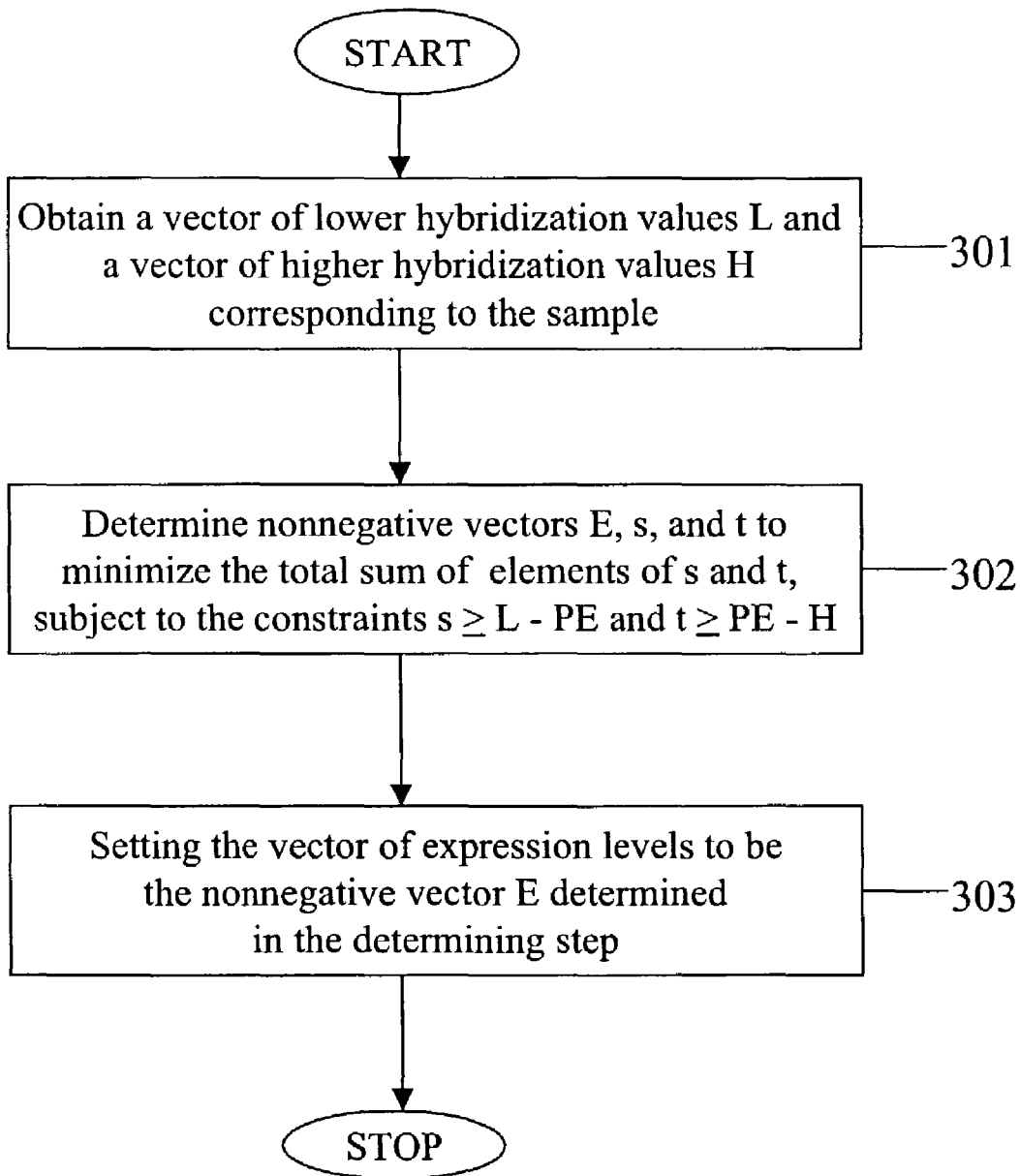


FIG. 3

METHOD FOR DETERMINING BIOLOGICAL EXPRESSION LEVELS BY LINEAR PROGRAMMING

BACKGROUND OF THE INVENTION

FIELD OF THE INVENTION

[0001] The present invention relates generally to systems and methods for the determination of biological expression levels present in a biological sample using data obtained from biological expression arrays.

[0002] The present invention includes the use of various technologies described in the references identified in the following LIST OF REFERENCES by the author(s) and year of publication and cross-referenced throughout the specification by reference to the respective number, in parentheses, of the reference:

LIST OF REFERENCES

- [0003] [1] BERTSEKAS, D. P. *Nonlinear Programming, 2nd Edition*. Athena Scientific, 1999.
- [0004] [2] BRAS, M., CLOAREC, J., BESSUEILLE, F., SOUTEYRAND, E., MARTIN, J., AND CHAUVET, J. Control of immobilization and hybridization on DNA chips by fluorescence spectroscopy. *Journal of Fluorescence* 10 (2000), 247-253.
- [0005] [3] CANTOR, C. R., AND SMITH, C. L. *Genomics*. Wiley, 1999.
- [0006] [4] CLARK, T., LEE, S., SCOTT, L. R., AND WANG, S. M. Computational analysis of gene identification with SAGE. *J. Comp. Bio.*, accepted.
- [0007] [5] HERNAIZ, M., LIU, J., ROSENBERG, R., AND LINHARDT, R. Enzymatic modification of heparan sulfate on a biochip promotes its interaction with antithrombin iii. *Biochemical and Biophysical Research Communications* 276 (2000), 292-297.
- [0008] [6] KANE, M. D., JATKOE, T. A., STUMPF, C. R., LU, J., THOMAS, J. D., AND MADORE, S. J. Assessment of the sensitivity and specificity of oligonucleotide (50 mer) arrays. *Nuclei Acids Research* 28, 22 (2000), 4552-4557.
- [0009] [7] LI, C., AND WONG, W. H. Model-based analysis of oligonecleotide arrays: expression index computation and outlier detection. *Proc. Natl. Acad. Sci. U.S.A.* 98 (2001), 31-36.
- [0010] [8] LI, C., AND WONG, W. H. Model-based analysis of oligonecleotide arrays: model validation, design issues and standard error application. *Genome Biology* 2 (2001), 32.1-32.11
- [0011] [9] LIPSHUTZ, R. J., FODOR, S. P. A., GINGERAS, T. R., AND LOCKHART, D. J. High density synthetic oligonucleotide arrays. *Nature Genetics* 21 (1999), 20-24.
- [0012] [10] LUEKING, A., HORN, M., EICKHOFF, H., BUSSOW, K., LEHRACH, H., AND WALTER, G. Protein microarrays for gene expression and antibody screening. *Analytical Biochemistry* 270 (1999), 103-111.
- [0013] [11] MACBEATH, G., KOEHLER, A. N., AND SCHREIBER, S. L. Printing small molecules as microarrays and detecting protein-ligand interactions en masse. *J. Am. Chem. Soc.* 121 (1999), 7967-7968.
- [0014] [12] MACBEATH, G., AND SCHREIBER, S. L., Printing proteins as microarrays for high-throughput function determination. *Science* 289, 5485 (2000), 1760-1763.
- [0015] [13] Walter, et al., S. L. Protein arrays for gene expression and molecular interaction screening. *Curr. Opin. Microbiol.* 3 (2000), 298-302.
- [0016] [14] MARK SCHENA, E. *Microarray biochip technology*. Eaton Publishing, 2000.
- [0017] [15] NOCEDAL, J., AND WRIGHT, S. J. *Numerical optimization*. Springer, 1999.
- [0018] [16] PEYRET, N., SENEVIRATNE, P. A., ALLAWI, H. T., AND SANTALUCIA, JR., J. Nearest-neighbor thermodynamics and NMR of DNA sequences with A-A, C-C, G-G, and T-T mismatches. *Biochemistry* 38 (1999), 3468-3477.
- [0019] [17] PILARSKY, C. P., SCHMITT, A. O., DAHL, E., AND ROSENTHAL, A. Microarrays—chances and challenges. *Curr. Opin. Molecular Therapeutics* 1 (1999), 727-737.
- [0020] [18] SCHENA, M., SHALON, D., DAVIS, R. W., AND BROWN, P. O. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science* 270 (1995), 467-470.
- [0021] [19] UETZ, P., GIOT, L., CAGNEY, G., MANSFIELD, T. A., JUDSON, R. S., KNIGHT, J. R., LOCKSHON, D., NARAYAN, V., SRINIVASAN, M., POCHART, P., QURESHI-EMILI, A., LI, Y., GODWIN, B., CONOVER, D., KALBFLEISCH, T., VIJAYADAMODAR, G., YANG, M., JOHNSTON, M., FIELDS, S., AND ROTHBERG, J. M. A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae*. *Nature* 403 (2000), 623-627.
- [0022] [20] VO-DINH, T., AND CULLUM, B. Biosensors and biochips: advances in biological and medical diagnostics. *Fresenius Journal of Analytical Chemistry* 366 (2000), 540-551.
- [0023] [21] WARRINGTON, J. A., DEE, S., AND TRULSON, M. Large-scale genomic analysis Affymetrix GeneChip (R) probe arrays. In *Microarray biochip technology*. Eaton Publishing, 2000, pp. 119-148.
- [0024] [22] WRIGHT, G., CAZARES, L., LEUNG, S., NASIM, S., ADAM, B., YIP, T., SCHELLHAMMER, P. F., GONG, L., AND VLAHOU, A. Protein-chip(R) surface enhanced laser desorption/ionization (SELDI) mass spectrometry: a novel protein biochip technology for detection of prostate cancer biomar-

kers in complex protein mixtures screening. *Prostate Cancer and Prostatic Diseases* 2 (1999), 264-276.

[0025] [23] YERSHOV, G., BARSKY, V., BELGOVSKIY, A., KIRILLOV, E., KREINDLIN, E., IVANOV, I., PARINOV, S., GUSCHIN, D., DROBISHEV, A., DUBILEY, S., AND MIRZABEKOV, A. DNA analysis and diagnostics on oligonucleotide microchips. *Proc. Natl. Acad. Sci. U.S.A.* 93 (1996), 4913-4918.

[0026] [24] LEE, S., CLARK, T., CHEN, J., ZHOU, G., SCOTT, L. R., ROWLEY, J. D., AND WANG S. M. Correct Identification of Genes from Serial Analysis of Gene Expression Tag Sequences. *GENOMICS Vol. 79. Number 4*, (April 2002), 598-602.

[0027] [25] SAGEMAP: SERIAL ANALYSIS OF GENE EXPRESSION TAG TO GENE MAPPING. National Center for Biotechnology Information (NCBI). www.ncbi.nlm.nih.gov/SAGE/.

[0028] [26] VAN DAM, R. M., AND QUAKE, S. R. Gene Expression Analysis with Universal n-mer Arrays, *Cold Spring Harbor Laboratory Press ISSN*. (2002), 145-152.

[0029] [27] ZHANG, L., ZHOU, W., VELCULESCU, V. E., KERN, S. E., HRUBAN, R. H., HAMILTON, S. R., VOGELSTEIN, B., AND KINZLER, K. W. Gene expression profiles in normal and cancer cells, *Science* 276 (1997), 1268-1272.

[0030] The entire contents of each reference listed in the LIST OF REFERENCES, are incorporated herein by reference.

DISCUSSION OF THE BACKGROUND

[0031] Biological expression arrays are introducing a type of massively parallel processing in experimental biology and medicine [17][20]. Gene expression arrays based on cDNA's [18] and on oligonucleotides [21] have reached a level in which the technology is well established, with books devoted to the subject [14]. Current research includes efforts to refine the technology [2].

[0032] More recently, protein expression arrays have been developed [12][13][22][19], and utilized to identify antibodies [10]. Other examples of biological expression arrays include small molecule expression arrays [11] as well as other specialized arrays [5].

[0033] Biological expression arrays are arrays of small biochemical experiments, each of which can be different from the other. Each "dot" on the array contains a reactant, called a probe. The set of all probes can be tested against a sample, which presumably contains a set of so-called targets which are to be determined, together with their quantity, or "expression level." For gene expression arrays, the targets are genes or gene fragments (called expressed sequence tags, or ESTs). The "expression levels" could as well be ratios of other more basic quantities, such as the ratio of the observed value for a perfect match and a mismatch [7][8].

[0034] The process of matching probes and targets for gene expression arrays is called hybridization. If the hybridization is perfect complementarity, the product may rather be

called a perfect complement rather than a hybrid. But when there are some complementarity mismatches, as is the general case, it is appropriate to think of the probe and target as coming from different genes, so the word hybrid is more appropriate. There is a difference between levels or detection signals of hybridization values (the input data) and expression values (the desired answer). In some cases, these are in a one-to-one relationship, so both are often called "expression" values, but in general they are not related in a simple way.

[0035] Arrays based on oligonucleotide probes of various lengths have been widely (often) used. A string of n bases is referred to as an n-mer. In the literature, one finds data from using 8-mers [23], 25-mers [21], 50-mers [6], and even longer oligonucleotides probes are currently in use as discussed on web sites. DNA-arrays based on probes consisting of significantly longer sequences (a few hundred bases) when synthesized via template-dependent enzymatic reactions are often called cDNA arrays [18]. A positive hybridization signal from the array is assumed to mean a complementary match (or near match) of sequences between the targets and the probes.

[0036] One successful approach to gene expression determination is that adopted by Affymetrix, Inc. (Santa Clara, Calif.) [21]. In their technique, oligonucleotides from several parts (loci) of a known gene are used to define array probes. This redundancy insures a high degree of certainty in identifying the expression of that gene, allowing precise discriminative detection between perfect matching and mismatching of probes to hybrids. (One presumes that care is also taken to be sure that the resulting oligonucleotides are unique and do not also appear in other relevant, either known or unknown genes or ESTs, which, in general, is impossible to be absolutely certain about.) For each perfect-match probe, a "mis-match-probe" is used which differs from the perfect-match probe by one single base, typically chosen in the middle of the oligonucleotide. Thus there should be an expected relation [3][16] between the expression level for a perfect-match complementary probe and its corresponding mis-match-probe. If these conditions are met for most, or many, of the probes for a given gene, it is highly likely that gene has been expressed.

[0037] A drawback to this approach is that multiple probes are required to determine a given gene. This approach is considered "polygamous" in that there are multiple probes presented on the array for a single target detection. When the expected relationship between the hybridization levels detected for a probe and its corresponding mis-match-probe are not met, the data for these probes is regarded as noise. There are many potential causes of such "noise" in expression data. For example, if some genetic contaminant is present that complementary matches exactly the mis-match-probe, the corresponding expression level for the "probe" (i.e., the perfect-match probe) would be of a "mis-match" level, inverting the signal ratio.

[0038] It would be advantageous to allow for more complex scenarios in which all of the discrete probe data is used for expression determination, rather than segregating them into groups and thus limiting the data from a group of discrete values to single-valued indications for a particular gene. This approach would also reduce the need for the probes to be "unique" markers for particular genes.

[0039] It could be possible to interrogate more genes even than the number of available probes, although this would not be possible if all genes were present in a given experiment. This would be done by having a particular probe to indicate the presence of multiple targets, albeit ambiguously. The new approach is “promiscuous” in that a single probe can indicate presence of multiple targets. In the promiscuous approach, all hybridization data becomes valuable signal. Complex hybridization scenarios (e.g., multiple-base mismatches) can (potentially) be included productively in the array hybridization data (signal) analysis.

[0040] Serial Analysis of Gene Expression, or SAGE™, is a technique designed to take advantage of high-throughput sequencing technology to obtain a quantitative profile of cellular gene expression [24][27][25][4]. A more detailed description of SAGE is given in U.S. Pat. No. 5,695,937, the entire contents of which are incorporated herein by reference. SAGE allows for the simultaneous quantitative analysis of a large number of mRNA transcripts. The SAGE method has two steps. First, short sequence tags (10-14 bp) are generated from the mRNA. Each tag should contain sufficient information to identify a unique transcript, provided that the tag is derived from a defined location within that transcript. Second, transcript tags are concatenated into a single molecule and then sequenced, revealing the identity of multiple tags simultaneously. The expression pattern of any population of transcripts can be quantitatively evaluated by determining the abundance of individual tags and identifying the gene corresponding to such a tag. The data produced by the SAGE method is a list of tags, with their corresponding count values, and can be portrayed as a digital representation of cellular gene expression.

SUMMARY OF THE INVENTION

[0041] Accordingly, it is an object of the present invention to provide a method and computer program product for determining biological expression levels from complex, “promiscuous” array hybridization data that also works well in the “polygamous” case.

[0042] Another object of the present invention is to provide a method and computer program product for determining biological expression levels from data derived from SAGE analysis.

[0043] Another object of the present invention is to provide a method and computer program product for determining biological expression levels from uncertain or noisy probe array hybridization data.

[0044] The above and other objects are achieved according to the present invention by providing a method for determining a matrix of expression levels corresponding to a set of biological targets and a set of biological samples, including obtaining a matrix of signal values P corresponding to the set of biological targets; computing a vector of expression levels for a sample in the set of biological samples using the matrix of signal values P ; storing the vector of expression levels computed in the computing step in a storage matrix; repeating the computing and storing steps for each sample in the set of biological samples; and outputting the storage matrix as the matrix of expression levels.

[0045] According to this method, the computing step includes obtaining a vector of signal values A corresponding

to the sample; determining a nonnegative vector E , a nonnegative vector s , and a nonnegative vector t that minimize a total sum of all elements of s and t , and satisfy a constraint $PE+s-t=A$; and setting the vector of expression levels to be the nonnegative vector E determined in the determining step.

[0046] In addition, further steps are provided for computing the vector of expression levels when the signal data is uncertain or noisy, including obtaining a vector of lower signal values L and a vector of higher signal values H corresponding to the sample, each element of L being less than or equal to a respective element of H ; determining a nonnegative vector E , a nonnegative vector s , and a nonnegative vector t , that minimize a total sum of all elements of s and t , and satisfy constraints $s \geq L-PE$ and $t \geq PE-H$; and setting the vector of expression levels to be the nonnegative vector E determined in the determining step.

[0047] An important aspect of the present invention is the formulation of the biological expression determination problem as a problem that can be solved using linear programming techniques.

[0048] Further, methods are provided for expression identification. In this problem, a particular target g gives rise to a certain array of signal values $P(g)$. Suppose a set U denotes a “universe” of targets. In any expression array experiment, some set of targets S , which is a subset of U , will be active, and are to be identified.

[0049] To that end, according to an aspect of the present invention, there is provided a method for identifying a set of biological targets $S=S(A)$ consistent with a vector of nonnegative signal values A , wherein the array $P(g)$ is compared to the vector A , for each target g in the set of universal targets U .

[0050] According to another aspect of the present invention, there is further provided a method for computing a multiplicity vector $D(A)$ that corresponds to the vector of nonnegative signal values A . Each element of $D(A)$ indicates the number of targets with a nonnegative signal value in the corresponding position in $P(g)$. This method makes use of the set $S(A)$ to compute the multiplicity vector $D(A)$.

[0051] According to another aspect of the present invention, there is further provided a method for identifying the subset of ambiguous targets $\Delta S(A)$ that may be expressed in a vector of nonnegative signal values A , but cannot be identified with certainty. This method makes use of both the set $S(A)$ and the multiplicity vector $D(A)$.

[0052] According to another aspect of the present invention, there is further provided a method for identifying the set of uniquely expressed biological targets $S(A) \setminus \Delta S(A)$ represented by a vector of nonnegative signal values A . This set consists of those targets in $S(A)$ that are not in the set $\Delta S(A)$.

BRIEF DESCRIPTION OF THE DRAWINGS

[0053] A more complete appreciation of the invention and many of the attendant advantages thereof will be readily obtained as the same becomes better understood by reference to the following detailed description when considered in connection with the accompanying drawings, wherein:

[0054] FIG. 1 is a flowchart showing the steps of determining a matrix of biological expression levels corresponding to a set of biological targets and a set of biological samples according to a first embodiment of the present invention;

[0055] FIG. 2 is a flowchart showing the steps of determining a vector of expression levels for a biological sample according to the first embodiment of the present invention; and

[0056] FIG. 3 is a flowchart showing the steps of determining a vector of expression levels for a biological sample according to a second embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENT

[0057] Biological arrays have as their digital output an array $A=(A_i)$ of hybridization values. Even though these often appear in a two dimensional array, it is assumed they are numbered by a single index i . A typical size array today has from a few thousand to as many as three-hundred thousand cells, and this could be expected to grow in the future, up to a million cells [9].

[0058] In a simplified model, these values are either zero or one (expressed or non-expressed: this might be achieved by thresholding real hybridization values). If such a determination could be made unambiguously, it would provide useful answers to biological questions. At the moment, even this is not a simple matter from a practical point of view. What is the mathematical structure of such a problem? In such a simplified system, particular targets g would cause a certain hybridization array $P(g)$ to occur. $P(g)$ is an array of hybridization levels, but again we can think initially of these as being either zero or one. This simplification of hybridization levels is sufficient to define the identification problem.

[0059] To begin with, assume that there is a set U of targets that denotes a "universe" of targets. The true "universe" may not be completely known at the moment, e.g., for gene expression arrays, since not all genes have been identified. But it is important to have this set as an underlying concept. In any expression-array experiment, some set S of targets that is a subset of U will be active, and it is this set that we want to identify. If, for some reason, it is known that the size of U can be restricted in a given experiment, then that can be done. The data from the experiment will be a particular hybridization array A of values of zero or one for each given probe. The identification problem can then be described as follows:

[0060] Determine the set $S \subset U$ of targets defined by

$$S = S(A) = \{g \in U : P(g) \leq A\} \quad (1)$$

[0061] where for two arrays A and B we say $B \leq A$ if A_i is one whenever B_i is one (equivalently $B_i \leq A_i$ for all i).

[0062] For the human genome, the size of U is expected to be about one-hundred thousand (10^5). It is harder to predict the expected size of a typical S , but an estimate of 10^3 to 10^4 might be reasonable.

[0063] A solution $S(A)$ to (1) can be computed by the following algorithm.

[0064] Algorithm 1 Loop over all $g \in U$ and test whether or not $P(g) \leq A$. If it is, include g in $S(A)$.

[0065] Each test $P(g) \leq A$ requires a number of operations proportional to the number of non-zeros in $P(g)$. Thus we have the following result.

[0066] Theorem 1 Algorithm 1 finds a solution to equation (1) in an amount of time that is linearly proportional to the size of the universe of targets U , and the constant of proportionality depends only on the average number of non-zeros in any $P(g)$.

[0067] The notion of $P(g)$ can be generalized to apply to a set: $P(S)_i = 1$ if $P(g)_i = 1$ for some $g \in S$, and $P(S)_i = 0$ if $P(g)_i = 0$ for all $g \in S$. Note that for any array A and any set $S \subset U$, we have $P(S) \leq A$ if and only if $P(g) \leq A$ for all $g \in S$. In general, if $R \subset S$ then $P(R) \leq P(S)$.

[0068] Algorithm 1 always constructs $S(A)$ so that $P(S(A)) \leq A$. One difficulty that can arise is that $P(S(A)) \neq A$. If this occurs, either there was an error in the experiment, or there was a target that was not represented in U . However, the following theorem says that the set $S(A)$ constructed in Algorithm 1 is the largest solution to the identification problem.

[0069] Theorem 2. Suppose there is a set $R \subset U$ such that $P(R) \leq A$. Then $R \subset S(A)$.

[0070] Proof: Suppose that $g \in R$. Since $P(R) \leq A$, we have $P(g) \leq A$. Hence by definition $g \in S(A)$.

[0071] Another problem can arise due to a kind of non-uniqueness. It may be that there are smaller sets $R \subset S$ such that $P(R) = P(S)$. Any targets in the difference $S \setminus R$ are possibly being expressed, but they cannot be identified for sure. However, it is possible to quantify the amount of non-uniqueness, as follows.

[0072] In constructing $S(A)$, it is possible to keep track of the multiplicity of each array point i , by which we mean the number of times this point had a non-zero value in $P(g)$ for some $g \in S(A)$. This defines an array $D(A)$ of such values, as follows:

$$D(A)_i = \text{cardinality } \{g \in S(A) : P(g)_i = 1\}. \quad (2)$$

[0073] Given an array $D(A)$ of degree values for $S(A)$ define $D(A)^-$ to be the array obtained from $D(A)$ by decrementing each positive array value. Thus the non-zero array values in $D(A)^-$ correspond to array values in $D(A)$ that were two or more, that is, ones that are multiply expressed. Now define

$$\Delta S(A) = \{g \in S(A) : P(g) \leq D(A)^-\}. \quad (3)$$

[0074] Lemma 1 For any $g \in S(A) \setminus \Delta S(A)$ there is at least one array index i_g such that $P(g)_{i_g} = 1$ and for all other $g' \in S(A)$ (with $g \neq g'$) we have $P(g')_{i_g} = 0$.

[0075] Proof: For a given $g \in S(A)$, let i_1, \dots, i_k enumerate all of the array indices such that $P(g)_{i_i} = 1$; $k \geq 1$ by the definition of $S(A)$. In particular, this means that $D(A)_{i_i} \geq 1$ for $i = 1, \dots, k$. If the degree of i_i is more than one for all i (i.e., $D(A)_{i_i} \geq 2$ for $i = 1, \dots, k$), then $g \in \Delta S(A)$. If $g \in S(A) \setminus \Delta S(A)$, then for one of the i 's, it must be true that $D(A)_{i_i} = 1$, and this means that for no other $g' \in S(A)$ can $P(g')_{i_i} = 1$.

[0076] Theorem 3 For any array A and for any set $R \subset U$ such that $P(R) = P(S(A))$, then $S(A) \setminus \Delta S(A) \subset R$.

[0077] Proof: To begin with, Theorem 2 implies that it can be assumed that $R \subset S(A)$. By Lemma 1, if $g \in S(A) \setminus \Delta S(A)$ then there is an index l_g such that $P(g)_{l_g} = 1$ and $P(g')_{l_g} = 0$ for all $g' \in S(A)$ with $g \neq g'$, and a fortiori for all $g \neq g' \in R$.

[0078] If $g \notin R$, then $P(g')_{l_g} = 0$ for all $g' \in R$. This would imply that $P(R)_{l_g} = 0$. By Theorem 2, $P(S(A) \setminus \Delta S(A)) \leq P(S(A))_{l_g}$. Since $P(g)_{l_g} = 1$ implies $P(S(A) \setminus \Delta S(A))_{l_g} = 1$, it must be true that $P(R)_{l_g} = P(S(A))_{l_g} = 1$ by assumption. Since this would be a contradiction, it must be true that $g \in R$.

[0079] Corollary 1 $S(A)$ is the largest possible set of expressed targets consistent with a hybridization array A . $S(A) \setminus \Delta S(A)$ is the largest set of uniquely expressed targets represented by A . $\Delta S(A)$ contains all the ambiguous targets which may be expressed, but cannot be identified with certainty. If $\Delta S(A)$ is empty, the solution is unique.

[0080] Having a set $\Delta S(A)$ of ambiguous targets does not mean that expression levels cannot be determined correctly. It may be possible to distinguish them due to the fact that the numerical value of the expression levels is different for different targets. An algorithm for determining these is considered below.

[0081] The computation of the array $D(A)$ and the set $S(A)$ can be accomplished simultaneously by the following modification of Algorithm 1. Assume the initial construction of a database of indices l^e_1, \dots, l^e_k of all indices such that $P(g)_{l^e_i} = 1$ for all $g \in U$. Note that this can be done computationally based on knowledge of the complementary matches between targets and probes and does not require experimental determination.

[0082] Algorithm 2 Set all array values of $D(A)$ to zero. Loop over all $g \in U$. For all i such that $P(g)_{l^e_i} = 1$, test whether or not $A_{ij} = 1$. If it is, include g in $S(A)$ and increment $D(A)_{ij}$.

[0083] Once the computation of the array $D(A)$ and the set $S(A)$ is completed, the set $\Delta S(A)$ can be computed by the following algorithm.

[0084] Algorithm 3 Loop over all $g \in S(A)$. For all i such that $P(g)_{l^e_i} = 1$, test whether or not $D_{ij} \geq 2$. If all of them are, include g in $\Delta S(A)$.

[0085] Theorem 4 Algorithm 2 computes the array $D(A)$ defined in (2) in an amount of time that is linearly proportional to the size of the universe of targets U . Algorithm 3 computes $\Delta S(A)$ in an amount of time that is linearly proportional to the size of $S(A)$. The constants of proportionality depend only on the average number of non-zeroes in any $P(g)$.

[0086] Simple examples now presented will show how the algorithms presented above work in practice. If there is only one $g \in U$ such that $P(g)_1$ is non-zero for a given array index l , then $A_{lj} \neq 0$ implies that $g \in S(A) \setminus \Delta S(A)$. Since g is unique, it can be referred to as g_l . If this holds for all Q , then $\Delta S(A) = \emptyset$ for any hybridization array A , and the expression determination is always unique. Note that the number of such that $g = g_l$ for a fixed g can be greater than one, which is the polygamous case.

[0087] The next most complicated case would be when each hybridization index l has at most two targets g such that $P(g)_1$ is nonzero. For simplicity, also assume that the hybridization array $P(g)$ has exactly two non-zero entries per

g . In this case, there is a canonical way to number the targets and the hybridization indices that simplifies the expression presentation and analysis, which is described below.

[0088] Select one g and call it g_1 and correspondingly number one of the array locations by "1" so that $P(g_1)_1$ is nonzero. Let array index number 2 be the other l such that $P(g_1)_1$ is nonzero. Thus $P(g_1)_1$ is nonzero precisely for $i=1, 2$. Suppose there is another g such that $P(g)_2$ is non-zero. Call that target number 2. Thus by definition $P(g_2)_2$ is now nonzero. If there is another l such that $P(g_2)_1$ is non-zero, let this be called the 3-rd array index. Continuing in this way, the result is a sequence of targets such that $P(g_i)_i$ is non-zero precisely for $i=j, j+1$.

[0089] Of course the process could terminate in one of two ways. First of all, there may be no other g such that $P(g)_i$ is non-zero, so target $i-1$ is uniquely identified by the i -th hybridization signal value. In this case, the first $i-1$ targets can be determined from the first i hybridization signal values. Then numbering can start over with the remaining targets and hybridization signal values following the same algorithm.

[0090] In the second case, the second hybridization index l with non-zero $P(g_{i-1})_l$ may be previously numbered, in which case there is a cycle back into the current group. By construction, each hybridization index $i > 1$ already has two targets g (precisely g_i and g_{i-1}) with non-zero hybridization signal values. Since the assumption is that there are at most two targets g such that $P(g)_1$ is non-zero, when a non-zero $P(g_{i-1})_1$ is found which is previously numbered, it must be $l=1$. Again, the algorithm can be started over with a new g to form a new group. By the assumption on the bound of at most two targets per hybridization signal value, these groups will all be disjoint. This means that the matrix $P_{ij} = P(g_j)_i$ is a lower bi-diagonal matrix, except for an occasional "loop back" entry.

[0091] Note that array values and targets are numbered using the same numerical values such that P_{ij} is non-zero for just $j=i, i+1$ in the generic case. For simplicity, consider hybridization array data A that is zero at the beginning and end of each group as defined above. Then it is easy to identify which parts of U are in $S(A)$ and $\Delta S(A)$. The hybridization array A is made up of a number of connected intervals $[i, i+k]$ in which the hybridization values are non-zero, with zero values in between. The interval boundaries identify the targets in the set $S(A) \setminus \Delta S(A)$, since they can be determined uniquely. However, the targets corresponding to interval interiors must be in $\Delta S(A)$. Thus, whenever there is an interval of three or more consecutive non zero hybridization values, there will be non-determinism in the expression pattern. As discussed below, (non-binary) hybridization signal levels can disambiguate these expression patterns

[0092] Consider the situation in which the individual expression levels are not just binary values. Suppose that a probe array has a digital output presented as an array $A = (A_i)$ of hybridization signal values, which are non-negative numbers. Similarly, particular targets g will cause a certain hybridization array $P(g)$ to occur. $P(g)$ is an array of hybridization levels indexed by l , each entry $P(g)_l$ a non-negative number. It would be a reasonable assumption that the non-zero values $P(g)$ might have the same magnitude for all l . However, this appears not to be the case in some situations [7][8].

[0093] In particular, it is quite reasonable to assume that $P(g)$ will have nonzero values corresponding to hybridizations with base-pair mismatches. Hybridization kinetics for oligos is well understood in solution [3][16], but the details of hybridization at surfaces remains a problem of interest. The approach presented here allows expression determination even in the case when complex hybridization occurs.

[0094] The expression determination problem can then be described as follows:

[0095] Determine a set $\{e_g \geq 0: g \in U\}$ of target expression values defined by

$$\sum_{g \in U} e_g P(g)_l = A_l \forall l. \quad (4)$$

[0096] Using vector and matrix notation, this can be simplified. Define P to be the matrix indexed by array indices l and by $g \in U$ with value $P(g)_l$. That is

$$P_{l,g} = P(g)_l \quad (5)$$

[0097] Define E to be a vector with components e_g indexed by $g \in U$. Then the expression determination problem is as follows:

[0098] Find a vector E of gene expression values defined by “solving”

$$PE = A, E \geq 0 \quad (6)$$

[0099] where $E \geq 0$ means that $e_g \geq 0$ for all $g \in U$.

[0100] Unfortunately, there need be no “solution” to (6) for arbitrary arrays $A \geq 0$ due to the positivity constraint, $E \geq 0$. The domain space (e.g., the set of possible E 's) of the matrix P is the size n of the set U , and the range or image set (e.g., the set of possible A 's) is of size k of the array. In the polygamous approach discussed earlier, $k \gg n$, meaning that the system $PE = A$ is an over-determined system. A necessary condition for solutions to $PE = A$ to exist is that A be in the range of P , and this corresponds to $k - n$ constraints which must be satisfied.

[0101] There is a simple restriction on the matrix P that will occur later, but described now for clarity. If a column of P is identically zero, say the column indexed by a particular $g \in U$, then this would mean that $P(g)_l = 0$ for all l , which in turn means that there is no probe that identifies g . Clearly this means that predictions about expression levels of g can not be made. In some sense, this would mean that g is not in the “universe” of targets that can be probed. Thus, a natural condition to impose on P (or more precisely, on the “universe” U) is that no column of P is identically zero.

[0102] In the promiscuous approach discussed earlier, it is reasonable to assume that $n \gg k$. This means that the system $PE = A$ is an under-determined system. A sufficient condition for solutions to $PE = A$ to exist is that P has full rank. The set of solutions to $PE = A$ is either empty or else an affine set of high dimension (dimension $n - k$ in the case of P full rank, even higher when P is rank deficient). It is possible that none of the elements of this set satisfy the constraint $E \geq 0$. The following algorithm does not assume that P has full rank; solutions in this case exist only for data satisfying constraints similar to those in the over-determined case.

[0103] A preferred method of the present invention is based on linear programming (LP) to determine whether the feasibility problem (6) has a solution, and to find such a solution if one exists. The initial linear programming formulation introduces vectors s and t , each containing k artificial variables indexed by l . The formulation is as follows:

$$\min_{E, s, t} \sum_l s_l + t_l \text{ subject to} \quad (7)$$

$$(E, s, t) \geq 0, PE + s - t = A. \quad (8)$$

[0104] Note several points about this LP. First, given any guess E satisfying $E \geq 2 \cdot 0$, a feasible starting point can be constructed by setting

$$s_l = -\min((PE - A)_l, 0), t_l = \max((PE - A)_l, 0). \quad (9)$$

[0105] Second, the objective value is bounded below by 0, because of the constraints that all components of s and t remain nonnegative. Thus the LP is feasible and bounded, and so it has a solution. Third, note that any solution E of the original system (6) can be transformed into a solution of (7-8) by setting $s = t = 0$. Using this fact, a certificate of feasibility for (6) can be obtained by solving (7-8); if the solution of (7-8) results in a strictly positive optimal value, then there exists no solution to (6). On the other hand, if the solution of (7-8) yields an objective value of zero, a solution to (6) is obtained by simply taking the E component of the solution to the LP. Finally, note that no assumptions on P (or k or n) are needed. These facts are collected in the following result.

[0106] Theorem 5 For any P and A , the minimization problem (7-8) always has a solution and can be started at the feasible point given by (9) for any E . The minimum value in (7) is zero if and only if the E component of the minimizer is a solution of (6).

[0107] One can show that there exists a solution of (7-8) that has at most k nonzeros in the solution vector (E, s, t) , and that the simplex method will find such a solution.

[0108] If in fact (6) is infeasible, the solution of (7-8) yields a nonnegative vector E for which the inconsistency in the equations $PE = A$ is minimized in the 1-norm.

[0109] A reasonable choice for initial E is the array $E(A)$ defined by $E(A)_g = 1$ (or a predetermined positive value) if $g \in S(A) \setminus \Delta S(A)$ (see Theorem 7).

[0110] FIG. 1 lists the steps in the preferred method for determining a matrix of expression levels corresponding to a set of biological targets and a set of biological sample.

[0111] In step 101, a matrix of hybridization values P corresponding to the set of biological targets is obtained.

[0112] Next, in step 102, a vector of expression levels for a sample in the set of biological samples using the matrix of hybridization values P is computed.

[0113] In step 103, the vector of expression levels computed in the computing step is stored in a storage matrix.

[0114] In step 104, if expression levels have been computed for all of the samples in the set of biological samples, the method proceeds to step 105. Otherwise, steps 102 and 103 are repeated.

[0115] Finally, in step 105, the storage matrix is outputted as the matrix of expression levels.

[0116] FIG. 2 lists the steps in the preferred method for computing the vector of expression levels for a sample in the set of biological samples using the matrix of hybridization values P.

[0117] In step 201, a vector of hybridization values A corresponding to the sample is obtained.

[0118] Next, in step 202, a nonnegative vector E, a nonnegative vector s, and a nonnegative vector t that minimize a total sum of all elements of s and t, and satisfy a constraint $PE+s-t=A$ are determined. A linear programming algorithm, such as the simplex method, can be used at this step.

[0119] Finally, in step 203, the vector of expression levels is set to be the nonnegative vector E determined in step 202.

[0120] It is often the case that probes are duplicated on a single biological expression array as a way of detecting hybridization errors. In principle, all duplicate probes should have the same hybridization signal levels, but in practice they do not. Different protocols can be observed to deal with the multiple data. One approach would be to average the values, and another would be to "vote" on the best value. Either of these, or some other method, could be used to define a single hybridization signal value for each probe, and the previous algorithms can be used.

[0121] Consider a more general formulation of the problem of determining expression levels, which is applicable when the correct values of some of the components of A_i is uncertain. Suppose that instead of a precise value A_i , there is an interval $[L_i, H_i]$ that contains the value. In other words, accept E as a valid vector of expression levels if

$$L_i \leq \sum_{g \in U} e_g P(g)_i \leq H_i.$$

[0122] (If the precise value is known, simply set both L_i and H_i equal to A_i .) The formulation analogous to (7-8) is then

$$\min_{E,s,t} \sum_i s_i + t_i \text{ subject to} \quad (10)$$

$$(E, s, t) \geq 0, s \geq L-PE, t \geq PE-H. \quad (11)$$

[0123] This formulation would in general be no more difficult to solve than (7-8).

[0124] FIG. 3 lists the steps in a method for computing the vector of expression levels for a sample in the set of biological samples using the matrix of hybridization values P, according to a second embodiment of the present invention.

[0125] In step 301, a vector of lower hybridization values L and a vector of higher hybridization values H is obtained by testing the biological sample with an array of reactive probes, each element of L being less than or equal to a respective element of H.

[0126] In step 302, a nonnegative vector E, a nonnegative vector s, and a nonnegative vector t, that minimize a total sum of all elements of s and t, and satisfy constraints $s \geq L-PE$ and $t \geq PE-H$ are determined. A linear programming algorithm, such as the simplex method, can be used at this step.

[0127] Finally, in step 303, the vector of expression levels is set to be the nonnegative vector E determined in step 302.

[0128] Returning to the formulation (7), consider the case in which the set (6) is nonempty. As mentioned above, the E component of any solution of (7-8) that has $s=t=0$ is a member of this set. In general, the set of vectors E satisfying (6) will be a (possibly unbounded) polytope. It may, however, be of interest to obtain some particular solution from the optimal set, or to learn something about the properties of this set.

[0129] To begin with, observe that in this case, the set of vectors E satisfying (6) will be a bounded polytope.

[0130] Theorem 6 Suppose that the matrix P has nonnegative entries and that none of its columns are identically zero. Then the set (6) is bounded.

[0131] Proof. The set (6) is unbounded if and only if there is a vector $F=[f_g]_{g \in U}$ such that

$$F \geq 0, PF=0, F \neq 0. \quad (12)$$

[0132] Suppose some component f_g of F is strictly positive. By the assumptions, there is an index l such that $P(g)_l$ is strictly positive. The l-th component of PF is at least $P(g)_l f_g$, which is strictly positive, contradicting $PF=0$. Hence no such component f_g exists, and $F=0$. Thus, no solution to (12) exists, and hence the set (6) is bounded.

[0133] It may be useful to find the solution of minimum-norm in case there exist multiple solutions. One can minimize the 1-norm by solving the following LP:

$$\min \sum_{g \in U} e_g \text{ subject to } E \geq 0, PE = A, \quad (13)$$

[0134] using the E component of the solution obtained from (7-8) as a feasible starting point for (13). Using a Euclidean norm criterion, one can solve

$$\min \frac{1}{2} \sum_{g \in U} e_g \text{ subject to } E \geq 0, PE = A. \quad (14)$$

[0135] This is a convex quadratic program [1] rather than a linear program, but it can still be solved with interior-point software (see Chapter 8, "Primal-dual interior-point methods" in *Numerical Optimization* by Nocedal and Wright [15]). Variants of (13) and (14) that use weighted norms or the α -norm can be formulated easily.

[0136] One could also seek extreme points of the set (6) by solving linear programs with random linear objectives. For instance the problem

$$\min \sum_{g \in U} x_g e_g \text{ subject to } E \geq 0, PE = A, \quad (15)$$

[0137] where each x_g is a random variable (drawn from some distribution that allows both positive and negative

values) will yield a vertex of the optimal polytope. By solving a sequence of problems of the form (15), for different choices of x , one can construct a subset of the optimal polytope by taking the convex hull of the solutions obtained in this process. By inspecting this subset, one may for instance be able to restrict the ambiguity produced by the data to certain subspaces. Noting that certain components e_g of the solution are identical (or nearly identical) regardless of the choice of x , one might conclude that these components are well determined by the data.

[0138] There is an intriguing connection between the discrete model (the identification problem) and the continuous (expression-level determination) problem. The sets S and ΔS allow us to determine the active set for the minimization problem of equations (7-8). But first consider modified definitions suitable for the more general expression values being considered here.

[0139] Generalizing the definition in equation (1), define

$$S = S(A) = \{g \in U : P(g)_g > 0 \rightarrow A_1 > 0 \forall I\}. \quad (16)$$

[0140] The solution $S(A)$ to (16) can still be computed by a simple modification to Algorithm 1. Similarly, equation (2) can be replaced by the more general definition

$$D(A)_1 = \text{cardinality } \{g \in S(A) : P(g)_g > 0\} \quad (17)$$

[0141] while keeping the definition (3) of $\Delta S(A)$. These can be computed by simple modifications of Algorithms 2 and 3. With these new definitions, one obtains the following modification of Lemma 1.

[0142] Lemma 2 For any $g \in S(A) \setminus \Delta S(A)$ there is at least one array index I^g such that $P(g)_{I^g} > 0$ and for all other $g' \in S(A)$ (with $g \neq g'$), $P(g')_{I^g} > 0$.

[0143] Using this Lemma, one can now prove the following result which characterizes the active set for a minimization problem for solving (6). Note that the solution (E, s, t) to (7-8) satisfies $PE = \hat{A} = A - s + t$.

[0144] Theorem 7 Suppose there exists an expression level vector E such that

$$PE = A, E \geq 0. \quad (18)$$

[0145] Then

$$S(A) \setminus \Delta S(A) \subset \{g \in U \mid e_g > 0\} \subset S(A). \quad (19)$$

[0146] Proof. Consider first the right inclusion in (19). Suppose for contradiction that $e_g > 0$ for some $f \in U \setminus S(A)$. Then by definition of $S(A)$ there exists an index k such that

$$P(f)_k > 0 \text{ and } A_k = 0.$$

[0147] From (18), it follows that

$$0 = A_k = \sum_{h \in U} P(h)_k e_h \geq P(f)_k e_f > 0,$$

[0148] giving the desired contradiction.

[0149] To prove the left inclusion in (19), take an arbitrary $g \in S(A) \setminus \Delta S(A)$ and try to show that $e_g > 0$. By Lemma 2, there is an index l such that

$$P(g)_l > 0, \text{ while } P(h)_l = 0 \text{ for all } h \in S(A) \setminus \{g\}. \quad (20)$$

[0150] In particular, since $g \in S(A)$, it follows from (16) that

$$A_l > 0. \quad (21)$$

[0151] In addition, it is true that $e_h = 0$ for all $h \in U \setminus S(A)$, by the argument above. By (18) and (20), it follows that for this index l

$$\begin{aligned} 0 < A_l &= \sum_{h \in U} P(h)_l e_h \\ &= P(g)_l e_g + \sum_{h \in S(A) \setminus \{g\}} P(h)_l e_h + \sum_{h \in U \setminus S(A)} P(h)_l e_h \\ &= P(g)_l e_g \end{aligned}$$

[0152] Therefore $e_g > 0$ as required, proving the result.

[0153] This result says that, if there is a solution to (6), then any algorithm for finding it may be restricted by assuming that $e_g = 0$ for $g \notin S(A)$. That is, compute $S(A)$ first, and then start looking for a solution E "supported" in $S(A)$. Moreover, one can be sure that all e_g are positive for $g \in S(A) \setminus \Delta S(A)$. For example, restrict the minimization problem in equations (7-8) to $E \in S(A)$ only.

[0154] One question of interest is what the linear programming model does when there is a hybridization array that relates to a target expression pattern for a target not in the universe U . Suppose that there is some $g \notin U$ with hybridization pattern $B = P(g)$. If there is an hybridization pattern $A = PE^0$ perturbed by adding B to get an hybridization pattern $A = A^0 + B$, then the linear programming model (7-8) will produce an answer (E, s, t) with

$$PE + s - t = A = A^0 + B = PE^0 + B. \quad (22)$$

[0155] Thus the error $E - E^0$ satisfies the error equation

$$P(E - E^0) = B - s + t. \quad (23)$$

[0156] Similarly, it is useful to know what the linear programming model will produce if there is a perturbation, e.g., due to noise. But the formulation reduces to the same considerations in which B is interpreted as a perturbation due to noise.

[0157] The algorithm of equations (7-8) is a minimization problem, and the vector (E^0, s^0, t^0) is a feasible approximant, where $s^0 - t^0 = B$, that is

$$s^0 = \max(B_1, 0), t^0 = \min(B_1, 0). \quad (24)$$

[0158] Note that

$$PE^0 + s^0 - t^0 = A^0 + s^0 - t^0 = A^0 + B = A. \quad (25)$$

[0159] By the optimality condition of (7-8)

$$\|(s, t)\|_1 \leq \|(s^0, t^0)\|_1 = \|B\|_1. \quad (26)$$

[0160] Thus the role of s and t can be interpreted as making the hybridization $A = A^0 + B - s + t$ satisfy the implicit constraint needed so that the system (6) has a solution $E \geq 0$ with right hand side A . One bound on the size of s and t is that s and t need be no bigger than required simply to cancel B completely, but s and t can be smaller as well. Thus s and t provide a lower bound for the size of the (unknown) perturbation B .

[0161] In the case that the matrix P is invertible, this means that whatever the perturbation B there will be a unique expression E attributed to it. The only indicator of error is the size of s and t .

[0162] The lower bound (26) means that the size of $\|(s, t)\|_1$ is a conservative estimation of the size of the hybridization error. Thus it may underestimate the error, but it will never overestimate it. Otherwise said, if it is large, then there is definitely a large discrepancy in the data, and it should not be trusted.

[0163] Recall that the error $E-E^\circ$ satisfies the error equation (23), namely, $P(E-E^\circ)=B-s+t$. But then (26) implies that

$$\|B-s+t\|_1 \leq \|s, t\|_1 + \|B\|_1 \leq 2\|B\|_1. \quad (27)$$

[0164] In the case that P is invertible, this allows us to give a bound on the expression error $E-E^\circ$.

[0165] Previously, two examples of possible expression array matrices were presented. In the first example there is only one $g \in U$ such that $P(g)_l$ is non-zero for a given array index l , which is referred to as g_l . One can number the targets and probes in such a way that the expression matrix $P_{ij}=P(j)_i$ has a simple form. Let the first target index $j=1$ correspond to some $g \in U$ for which $P(g)_l$ is non-zero for some l . There is some number $I_1 \geq 1$ of array indices l such that $P(g)_l$ is non-zero. Number these array indices $1, \dots, I_1$. Now pick another $g \in U$ for which $P(g)_l$ is non-zero for $I_2 \geq 1$ array indices l , and call this target $j=2$ and number these array indices I_1+1, \dots, I_1+I_2 . Continuing in this fashion, one constructs a matrix P with only one non-zero per row, such that the j -th column consists of $I_1+I_2+\dots+I_{j-1}$ zeros, followed by I_j non-zeros, and then followed by all zeros. For simplicity, assume that all the non-zeros are ones.

[0166] As shown above, expression determination is always unique in this case. However, it is interesting to consider how the algorithm (7-8) deals with this case, especially in the presence of noisy data. The role of the extra variables s and t in this case is to make sure that the hybridization array $A-s+t$ is in the range of the matrix P . The range of P is easy to describe. It consists of vectors A such that the entries I_1+1, \dots, I_1+I_2 all have the same value, for each j (for completeness, define $I_0=0$). For any A not satisfying this constraint, algorithm (7-8) will adjust the variables s and t to make the vector $A-s+t$ satisfy this property.

[0167] It suffices to see what happens with a single block j , so consider the case when P is a $k=I_1$ by 1 matrix (one expression value only, with k hybridization array values). For simplicity, assume that the first $k-1$ hybridization values of A are $\alpha < 0$ and the k -th value is $\beta \leq \alpha$. This would correspond to a simple error in one of the hybridization array values. Then it is easy to see that the optimal vectors s and t will have the following form. The first $k-1$ values of s will be some value σ and the k -th value will be zero. The first $k-1$ values of t will be zero, and the k -th value will be some value τ . To have $A-s+t$ be in the range of P , it must be true that $\alpha - \sigma = \beta + \tau$. Thus

$$\Sigma_i (s_i + t_i) = \sigma(k-1) + \tau = \alpha(k-2) + \alpha - \beta. \quad (28)$$

[0168] If $k > 2$, this is minimized by taking $\sigma=0$ (and so $\tau=\alpha-\beta$) which corresponds to the expression value obtained by the "voting" algorithm: the consensus $k-1 > 1$ values are

confirmed. In the case $k=2$ (when there is a tie), the resulting solution adjusts the array so that $A-s+t$ corresponds to the average $(\alpha+\beta)/2$. Thus the optimization algorithm (7-8) does a very reasonable thing in this case.

[0169] When the data is more complex, the effect of the optimization algorithm (7-8) is more complicated to describe. The resulting "consensus value" will be one of the array values, the one that minimizes the L_1 norm of the deviation from the other values. FIG. 1 shows how this would work with some synthetically generated random data. As shown in FIG. 1, the algorithm does a good job of "healing" errors introduced by noise.

[0170] In the second example above the case in which there are at most two targets g such that $P(g)_l$ is non-zero for any l , and for which there are exactly two non-zero hybridization levels per target was considered. In this case, the expression matrix $P_{ij}=P(j)_i$ is a lower bi-diagonal matrix, except for an occasional "loop back" entry. Each of these "loop backs" marks a block of independent expression, so focus on just one. For simplicity, just assume that $P=(p_{ij})$ is of the form $p_{11}=1$, $p_{i,i-1}=p_{i,i}=1$ for all $i=2, \dots, n$ and the rest are zero. Expression analysis in this case is equivalent to solving the equation $PE=A$, and it is possible to invert the matrix P explicitly. Let $Q=(q_{ij})$ be the lower-triangular matrix whose i -th row satisfies $q_{i,j}=(-1)^{i-j}$. Then Q is the inverse of P . Thus expression levels can be determined from $E=QA$, provided these values are non-negative.

[0171] One thing this example makes clear is that the ambiguity resulting from $\Delta S(A) \neq \emptyset$ above, which occurs for any array data with three or more consecutive non-zero values, does not lead to an inability to determine expression levels. Any hybridization array A will yield an unambiguous expression $E=QA$. Thus utilizing (non-binary) expression levels leads to a more robust identification system.

[0172] Having an explicit inverse for P allows one to study the effect of errors in the data. Suppose an array A is perturbed by $\epsilon=(\epsilon_i)$. The resulting expression values are given by $\hat{E}=Q(A+\epsilon)=E+Q\epsilon$, provided these values are positive. Suppose that $\epsilon_i=\delta(-1)^{i-1}$. Then $(Q\epsilon)_i=\delta_i$. This means that the error can be as large as the number of expression values times the perturbation. Otherwise said, the inverse matrix Q can amplify the error significantly.

[0173] One problem with the current approach is the limitation to two non-zero hybridization array locations. It is certainly possible to pick oligos which can target an essentially arbitrary number of genes [4]. There is another simple expression matrix one can examine. Suppose that the i -th probe identifies the first i targets. This corresponds to assuming that $p_{ij}=1$ for all $j=1, \dots, i$ for $i=1, \dots, n$. In this case the inverse matrix Q has the simple form $q_{11}=1$, $q_{i,i}=-1$, and $q_{i,i-1}=1$ for all $i=2, \dots, n$.

[0174] In this example, it is possible to characterize the arrays A for which the corresponding expression levels E are non-negative. The condition $E=QA \geq 0$ can be used inductively as follows. First of all, $0 \leq e_1 = \alpha_1$, but this only says that the hybridization array value α_1 should not be negative. But for $i > 1$, $0 \leq e_i = \alpha_i - \alpha_{i-1}$, and this says that the array values should not decrease in this ordering of the values. This is reasonable since each succeeding value represents more and more targets.

[0175] Previously, an analysis of what the linear programming model does when there is an array $A=PE^\circ+B$ for some

perturbation B was considered. The algorithm (7-8) produces a solution (E, s, t) where

$$PE=A-s+t. \quad (29)$$

[0176] Then the role of s and t can be interpreted as making the vector $A-s+t$ satisfy the constraint needed so that $E \geq 0$. In the example, this means that $A-s+t$ must be non-decreasing; recall that $A-B=PE^0$ must be non-decreasing to begin with, but there is no reason that A would be.

[0177] Consider now the situation when targets in the same experiment have been labeled in different ways, e.g., so the targets appear on the array with different colors. For example, targets could be from different samples: a "normal" sample might be labeled with a green fluorescence label and another sample might be labeled with a red label. If both types of samples are present in equal amounts, the resulting color will appear yellow.

[0178] From a mathematical point of view, the expression levels are simply vectors, with one component for each color. Let us assume that there is some number $c \geq 1$ of colors. In the more general, multi-color case, one supposes that the probe array has a digital output presented as an array $A=(A_i)$ of hybridization vectors, where each component A_i^i is a non-negative number, for $i=1, \dots, c$.

[0179] Particular targets g will still generate a hybridization array P(g) of scalar hybridization values for each l, again with non-negative components. The color would be determined by the marking, but the response would presumably be the same independent of the marking color, or at least that is the assumption. Let an expression vector $e_g \geq 0$ if $e_g^i \geq 0$ for all $i=1, \dots, c$.

[0180] The multi-color expression determination problem is then as follows:

[0181] Determine a set $\{e_g \geq 0: g \in U\}$ of gene expression vectors defined by "solving"

$$\sum_{g \in U} e_g P(g)_l = A_l \quad \forall l. \quad (30)$$

[0182] Using vector and matrix notation, this can be simplified as before. Define P to be the matrix indexed by array indices l with value P(g)_l and by $g \in U$. Define E to be a vector-valued array indexed by $g \in U$, and define $E \geq 0$ to mean that $e_g \geq 0$ for all $g \in U$. Then the multi-color expression determination problem is as follows:

[0183] Determine an array $E \geq 0$ of gene expression vectors defined by "solving"

$$PE=A. \quad (31)$$

[0184] This is really just c independent problems, of the form

$$PE^i=A_i, \quad E_i \geq 0. \quad (32)$$

[0185] for $i=1, \dots, c$. Thus the techniques and results discussed above apply. If the original data for the problem is an array A of color values, these will have to be decomposed into constituent colors A_i , $i=1, \dots, c$. But after that, the problem can be solved by techniques developed for the single color case.

[0186] Algorithms have been presented based on linear programming that determine expression values from arrays of biological experiments with complex relationships between probes and targets. The algorithms have been analyzed abstractly and bounds have been given to relate certain computed quantities to hybridization error. They have been shown to work for both "promiscuous" array data, in which there may be multiple targets indicated by a single probe, as well as in the "polygamous" case where there are multiple probes for a single target.

[0187] In an alternative embodiment, the method for determining biological expression levels can be used with Serial Analysis of Gene Expression (SAGE™) data [4][24][25][27]. In standard SAGE analysis, probes are represented by SAGE tags, n-mer sequences following a given marker sequence, such as CATG. The targets are the genes that contain them. Of course, there can be many such genes [24][4]. Using the notation described above, the corresponding matrix entry P(g)_l is non-zero if the l-th SAGE tag is in the gene (or EST) g, and zero otherwise. For simplicity, one can take the non-zero entries to be equal to one. Multiple matches correspond to a row of P with multiple non-zeros. However, in any given column, there will be at most one non-zero entry. Thus, there is no way to distinguish different expression levels with standard SAGE analysis using the method of the present invention described above.

[0188] However, if multiple SAGE analyses are performed with multiple markers (e.g., CTAG, etc.), one can get multiple non-zero entries in columns of P. This provides a set of equations that are amenable to the method of the present invention. Note that there is no concept of mis-hybridization with SAGE (the error rates in sequencing are very small). However, it might be that different markers (CTAG versus CATG) would have different affinity levels, leading to different non-zero coefficients in P. Note that in multi-SAGE analysis, the index l would be different for different markers. That is, CATGAACCGGTTAA is different from CTAGAACCGGTTAA. There are potentially sixteen different 4-mer palindromes available to use as markers in multi-SAGE analysis. This would lead to having up to sixteen non-zero elements in each column of the matrix P.

[0189] It will be appreciated from the foregoing that the present invention represents a significant advance over other systems and methods for determining biological expression levels. It will also be appreciated that, although a limited number of embodiments of the invention have been described in detail for purposes of illustration, various modifications may be made without departing from the spirit and scope of the invention. Accordingly, the invention should not be limited except as by the appended claims.

1. A method for determining a matrix of expression levels corresponding to a set of biological targets and a set of biological samples, comprising:

obtaining a matrix of signal values P corresponding to the set of biological targets;

computing a vector of expression levels for a sample in the set of biological samples using the matrix of signal values P;

storing the vector of expression levels computed in the computing step in a storage matrix;

- repeating the computing and storing steps for each sample in the set of biological samples; and
- outputting the storage matrix as the matrix of expression levels.
2. The method of claim 1, wherein the computing step comprises:
- obtaining a vector of signal values A corresponding to the sample;
- determining a nonnegative vector E, a nonnegative vector s, and a nonnegative vector t that minimize a total sum of all elements of s and t, and satisfy a constraint $PE+s-t=A$; and
- setting the vector of expression levels to be the nonnegative vector E determined in the determining step.
3. The method of claim 2, wherein obtaining the vector of signal values A comprises:
- testing the sample with an array of reactive probes.
4. The method of claim 2, wherein obtaining the vector of signal values A comprises:
- testing the sample with an array of reactive probes, at least one probe in the array of reactive probes being indicative of more than one target in the set of biological targets.
5. The method of claim 2, wherein obtaining the vector of signal values A comprises:
- testing the sample with an array of reactive probes, each probe in the array of reactive probes being a sequence of oligonucleotides.
6. The method of claim 2, wherein obtaining the vector of signal values A comprises:
- testing the sample with an array of reactive probes including at least one probe which is a sequence of oligonucleotides from at least one part of a known gene.
7. The method of claim 2, wherein the determining step comprises:
- identifying a set of biological targets $S(A)$ consistent with the vector of nonnegative signal values A; and
- setting, for each target in the set of biological targets that is not in $S(A)$, a respective element of the nonnegative vector E to zero.
8. The method of claim 2, wherein the determining step comprises:
- using a linear programming algorithm to determine the nonnegative vector E, the nonnegative vector s, and the nonnegative vector t.
9. The method of claim 8, wherein the determining step further comprises:
- constructing a feasible starting point for the linear programming algorithm, for an initial nonnegative vector E_0 , by initializing an l-th element of the vector s to be $s_l = -\min((PE_0 - A)_l, 0)$, and initializing an l-th element of the vector t to be $t_l = -\max((PE_0 - A)_l, 0)$.
10. The method of claim 9, wherein the determining step further comprises:
- constructing the initial nonnegative vector E_0 , by setting each element of E_0 to be either zero or, if a corresponding biological target could have uniquely contributed to the vector of signal values A, to a predetermined positive value.
11. The method of claim 2, wherein obtaining the vector of signal values A comprises:
- generating a plurality of short nucleotide sequence tags from the sample;
- concatenating the plurality of short nucleotide sequence tags into a single molecule;
- sequencing the molecule to determine a count for each tag in the plurality of short nucleotide sequence tags; and
- mapping the counts determined in the sequencing step into the vector of signal values A.
12. The method of claim 1, wherein the matrix of signal values P obtained in the obtaining step includes more columns than rows.
13. The method of claim 1, wherein obtaining the matrix of signal values P comprises:
- testing each target in the set of biological targets with an array of reactive probes.
14. The method of claim 1, wherein obtaining the matrix of signal values P comprises:
- testing each target in the set of biological targets with the array of reactive probes,
- each target being a gene or a gene fragment.
15. The method of claim 1, wherein the computing step comprises:
- obtaining a vector of lower signal values L and a vector of higher signal values H corresponding to the sample, each element of L being less than or equal to a respective element of H;
- determining a nonnegative vector E, a nonnegative vector s, and a nonnegative vector t, that minimize a total sum of all elements of s and t, and satisfy constraints $s \geq L - PE$ and $t \geq PE - H$; and
- setting the vector of expression levels to be the nonnegative vector E determined in the determining step.
16. The method of claim 1, wherein obtaining the matrix of signal values P comprises:
- generating a plurality of short nucleotide sequence tags for a target in the set of biological targets;
- concatenating the plurality of short nucleotide sequence tags into a single molecule;
- sequencing the molecule to determine a count for each tag in the plurality of short nucleotide sequence tags;
- mapping the counts determined in the sequencing step into the matrix of signal values P; and
- repeating the generating, concatenating, sequencing, and mapping steps for each target in the set of biological targets.
17. A method for determining a vector of expression levels, comprising:
- obtaining a vector of signal values A corresponding to a biological sample;
- obtaining a matrix of signal values P corresponding to a set of biological targets;

- determining all N nonnegative vectors satisfying an equation $PE=A$;
- outputting, if $N=1$, the nonnegative vector determined in the determining step as the vector of expression levels;
- computing, if $N=0$, a nonnegative vector E^* that minimizes an L_1 norm of an inconsistency in the equation $PE=A$, and outputting the nonnegative vector E^* as the vector of expression levels; and
- selecting and outputting, if $N>1$, one of the N nonnegative vectors determined in the determining step as the vector of expression levels.
- 18.** The method of claim 17, wherein the selecting and outputting step comprises:
- choosing one of the N nonnegative vectors determined in the determining step that minimizes a vector norm, the vector norm being a total sum of all elements.
- 19.** The method of claim 17, wherein the selecting and outputting step comprises:
- choosing one of the N nonnegative vectors determined in the determining step that minimizes a vector norm, the vector norm being a total sum of the squares of all elements.
- 20.** A method for identifying a set of biological targets $S(A)$ consistent with a vector of nonnegative signal values A , comprising:
- obtaining a vector of nonnegative signal values $P(g)$ for a target g in a known universe of biological targets U ;
- determining if, for every positive element of $P(g)$, whether a respective element of A is also positive, and if so, including the target g in the set of biological targets $S(A)$;
- repeating the obtaining and determining steps for each target g in the known universe of biological targets U ; and
- outputting the set of biological targets $S(A)$.
- 21.** A method for identifying a set of uniquely expressed biological targets consistent with a vector of nonnegative signal values A , comprising:
- identifying a set of targets $S(A)$ consistent with the vector of nonnegative signal values A ;
- identifying a set of ambiguous biological targets $DS(A)$ that may be expressed in the vector of nonnegative signal values A , but cannot be identified with certainty; and
- outputting those targets that are in $S(A)$, but not in $DS(A)$, as the set of uniquely expressed biological targets consistent with the vector of nonnegative signal values A .
- 22.** The method of claim 21, wherein identifying the set of ambiguous biological targets $DS(A)$ comprises:
- computing a multiplicity vector $D(A)$ corresponding to A ;
- obtaining a vector of nonnegative signal values $P(g)$ for a target g in $S(A)$;
- including the target g in $DS(A)$ if, for each positive element of $P(g)$, a respective element of $D(A)$ is at least two; and
- repeating the obtaining and including steps for every target g in $S(A)$.
- 23.** The method of claim 22, wherein computing the multiplicity vector $D(A)$ comprises:
- initializing $D(A)$ to all zeros;
- obtaining a vector of nonnegative signal values $P(g_o)$ for a target g_o in $S(A)$;
- incrementing by one those elements of $D(A)$ that correspond to positive elements of $P(g_o)$; and
- repeating the obtaining a vector of nonnegative signal values $P(g_o)$ step and the incrementing step for each target g_o in $S(A)$.
- 24.** A computer program product configured to store plural computer program instructions which, when executed by a computer, causes the computer to determine a matrix of expression levels corresponding to a set of biological targets and a set of biological samples, by performing plural steps comprising:
- obtaining a matrix of signal values P corresponding to the set of biological targets;
- computing a vector of expression levels for a sample in the set of biological samples using the matrix of signal values P ;
- storing the vector of expression levels computed in the computing step in a storage matrix;
- repeating the computing and storing steps for each sample in the set of biological samples; and
- outputting the storage matrix as the matrix of expression levels.
- 25.** The computer program product as claimed in claim 22, wherein the computing step comprises:
- obtaining a vector of signal values A corresponding to the sample;
- determining a nonnegative vector E , a nonnegative vector s , and a nonnegative vector t that minimize a total sum of all elements of s and t , and satisfy a constraint $PE+s-t=A$; and
- setting the vector of expression levels to be the nonnegative vector E determined by the preceding determining step.
- 26.** A system configured to determine a matrix of expression levels corresponding to a set of biological targets and a set of biological samples by performing the steps recited in any one of claims 1-16.
- 27.** A system configured to determine a vector of expression levels by performing the steps recited in any one of claims 17-19.