Error Propagation in Sparse Linear Systems with Peptide-Protein Incidence Matrices

Peter Damaschke and Leonid Molokov

Department of Computer Science and Engineering
Chalmers University, 41296 Göteborg, Sweden
[ptr,molokov]@chalmers.se

Abstract. We study the additive errors in solutions to systems $Ax = b$ of linear equations where vector b is corrupted, with a focus on systems where A is a 0,1-matrix with very sparse rows. We give a worst-case error bound in terms of an auxiliary LP, as well as graph-theoretic characterizations of the optimum of this error bound in the case of two variables per row. The LP solution indicates which measurements should be combined to minimize the additive error of any chosen variable. The results are applied to the problem of inferring the amounts of proteins in a mixture, given inaccurate measurements of the amounts of peptides after enzymatic digestion. Results on simulated data (but from real proteins split by trypsin) suggest that the errors of most variables blow up by very small factors only.

Keywords: protein mixture inference, linear system, error propagation, bipartite graph, shortest path

1 Introduction

Suppose that we are given an analytical chemistry problem that obeys the following informal "axioms".

- An unknown mixture of *compounds* is given.
- The compounds can be split into *constituents* by a certain chemical reaction triggered by an external compound E.
- The compounds react only with E and decay independently of each other. No other chemical reactions take place.
- We have a database telling which constituents result from the splitting of each compound.
- The compounds are hard to measure directly.
- The amount of each constituent can be measured directly.

Now we would like to infer which compounds are in the mixture, along with the amount of each one. Obviously this problem can be formulated as a system $Ax = b$ of linear equations, where A is the incidence matrix of constituents (one

per row) and compounds (one per column), vector b is the measured mixture of constituents, and x is the unknown vector of compounds.

Only nonnegative vectors make sense here: $b \geq 0$ and $x \geq 0$. Therefore we can immediately set to zero all variables that appear in rows j where $b_j = 0$, thus leaving a system $Ax = b$ with a (perhaps) smaller matrix A where all entries of b are strictly positive.

However, reality dictates that we have to take noise in account. We cannot measure b exactly. In the following we adopt the assumption that all entries of b have an additive error of at most some parameter value ϵ. Instead of the true b_j we therefore measure some amount between $b_j - \epsilon$ and $b_j + \epsilon$. This involves a certain simplification because one may be able to measure some constituents more accurately than others, however the assumption is not too artificial and allows for nice mathematical analysis, as we will see. Note also that one may disregard entries known to be unreliable right from the beginning and restrict the linear system to selected rows.

We may observe $b_j = 0$ instead of some true $b_j \leq \epsilon$, thus discarding some variables that should actually be positive. But since we have $x \geq 0$, affected variables are in this case bounded by ϵ, too, hence it is not a bad mistake to ignore them. In the following we are aim for upper bounds on the additive errors of the variables in general. Note that only the structure of matrix A matters.

Our aim is to classify variables by error bounds in a specific application where A has only entries 0 and 1, and A is sparse, i.e., has very few 1 entries: We are particularly interested in the question of absolute quantification of proteins in a protein mixture by means of shotgun proteomics. The general idea of shotgun proteomics is to split large protein molecules into smaller peptides, which later can be observed by mass spectrometry. That is, the compounds are proteins, and constituents are peptides after enzymatic digestion. The problem of just inferring which proteins are present leads to the combinatorial problem of Set Cover or, equivalently, Hitting Set. But we may also want to quantify proteins, e.g., in order to detect changes of protein expression. See, e.g., [11, 2, 13, 9] for more background.

Evidence of protein presence is established through analysis of peptide observations, as each protein is believed to be generating consistent peptide observations. Many proteins contain unique peptides, what means that after splitting they will produce some peptide that other proteins would not. Current quantification procedures use this fact and attempt to quantify proteins expression by measuring unique peptides' quantities with one or another lab technique, e.g., iTRAQ. In the system described in [3], relative or absolute quantification is done by mass spectrometry. Also error propagation is addressed. However it is said that for "protein quantification, only unique peptides are taken into consideration, whereas peptides belonging to more than one protein sequence are only used for proving the identification of the corresponding proteins" although they are ubiquitous.

A natural question is whether the linear system of equations could be used to include also ambiguous peptides directly in the inference process (assuming

bounded errors), and how measurement errors would then propagate. Intuitively, unique peptides' quantities will not affect noise level, while the more shared a peptide is, the more noisy are the inferred protein quantities. We stress that error propagation solely depends on the incidence matrices, and the present paper focuses on this aspect. In particular, this work is not concerned with measurement technologies but only with the structure of the protein-peptide incidence matrices. We observe that many peptides appear in only two candidate proteins, which also allows a graph-theoretic view of the problem.

Apparently the work most related to ours is "the first attempt to use shared peptides in protein quantification" made in [6]. Non-unique peptides can also contribute to quantification, specifically to quantify those proteins which either lack unique peptides themselves or unique peptide observations. The suggested LP model incorporates shared peptides' relative quantities and attempts to explain them by minimizing some error measure. Different detectability of peptides can be modelled by individual scaling factors applied to the entries of vector b before giving it to the linear system, whereas matrix A is not affected. While error minimization is quite natural, it might still not yield the proper prediction of actual quantitites. Despite similarities to [6], the present paper explicitly focuses on the aspect of controlled errors and their propagation and also makes some graph-theoretic contributions.

Not surprisingly, error bounds in linear systems is an extensively studied subject, see e.g. [1, 4, 12]. However, here we want elementary, easily applicable bounds for the case of 0, 1-matrices A with sparse rows and perturbations limited to b, and we aim at conclusions for the protein inference problem (using simulated digestion and mixture data, however from a real protein database.) A related problem is to explain the measurements by minimal mixtures (following the parsimony principle), or algebraically, finding and enumerating the solutions with minimal sets of nonzero variables in linear systems. In [5] we proved fixed-parameter tractability of that problem, parameterized by the solution size and the number of variables per equation. Another, less related problem with data mining applications is the reconstruction of a low-rank factorization of a given matrix under extra assumptions, as in [8] where an MILP formulation is used to attack the problem.

The paper is organized as follows. Section 2 gives the necessary definitions and basic facts. The central definition is a measure of error propagation as the solution to some linear program (LP). A relevant special case is linear systems where A is the incidence matrix of a graph. In Section 3 we obtain a complete characterization of the optimal error measure in this case. Section 4 briefly discusses the sparsity of optimal combinations of measures, and a method to obtain optimality certificates. After perturbation of vector b, a system $Ax = b$ may have no solution at all, and then we would like to satisfy all equations as good as possible. Section 5 gives a combinatorial characterization of the optimal solution in the case of bipartite graphs. (Due to space limitations we could not insert figures to illustrate the graph-theoretic results and proofs.) In Section 6 we report some simulation results and draw conclusions.

2 Error Bounds Through Linear Programming

Vectors x are always understood to be column vectors, therefore the transposed x^T is always a row vector. For two vectors x and y of equal lengths we mean by $x \leq y$ that $x_i \leq y_i$ for all i, similarly $x < y$ means that $x_i < y_i$ for all i. Symbol $|x|$ denotes a vector that consists of the entries $|x_i|$ for all i. The notation should not be confused with any norm of x. Instead, by $||.||_1$ we mean the ℓ_1-norm, i.e., the sum of absolute values of all entries of a vector. Symbol $\bar{1}$ denotes a vector where all entries are 1, with a length that will always be clear from context. Observe that the ℓ_1-norm can be written as the inner product $||x^T||_1 = |x|^T\bar{1}$. The incidence vector of a subset V of variables is the vector where all variables in V are set to 1, and all others are 0. A unit vector is an incidence vector of a single variable: $(0, \ldots, 0, 1, 0, \ldots, 0)$.

The following is the central concept of the paper. We first state it formally and discuss it later after having proved some properties.

Definition 1. *Let A be a fixed $m \times n$-matrix and z^T an n-vector. With respect to A we define $e(z^T) := \min\{||y^T||_1 : \ y^TA = z^T\}$. If no such y^T exists, $e(z^T)$ remains undefined.*

Theorem 1. *Consider an admissible linear system $Ax = b$ with $m \times n$-matrix A. We also denote by x a specific solution vector. Let z^T be any n-vector for which $e(z^T)$ is defined, and $\epsilon > 0$ some scalar. Furthermore let b' be an m-vector with $|b - b'| \leq \epsilon\bar{1}$. Then there exists an n-vector x' with $|Ax' - b'| \leq \epsilon\bar{1}$. Moreover, any such x' satisfies $|z^Tx' - z^Tx| \leq 2e(z^T)\epsilon$. If $Ax' = b'$ is admissible, too, i.e., has an exact solution x' then $|z^Tx' - z^Tx| \leq e(z^T)\epsilon$.*

Proof. The first assertion is trivial: In particular, $x' := x$ yields

$$|Ax' - b'| = |Ax - b'| \leq |b - b'| \leq \epsilon\bar{1}.$$

To show the second assertion, let y^T be a vector as in the definition of $e(z^T)$, that is, $y^TA = z^T$ and $e(z^T) = ||y^T||_1$. Then we get:

$$|z^Tx' - z^Tx| = |y^TAx' - y^Tb| = |y^T(Ax' - b)| = |y^T(Ax' - b' + b' - b)|$$

$$\leq |y^T(Ax' - b')| + |y^T(b' - b)| \leq |y|^T|Ax' - b'| + |y|^T|b' - b|$$

$$\leq 2|y|^T\epsilon\bar{1} = 2\epsilon|y|^T\bar{1} = 2\epsilon||y^T||_1 = 2e(z^T)\epsilon.$$

The third assertion follows from the same analysis, since one term disappears. □

In particular, if z^T is the incidence vector of a set V, Theorem 1 says that the uncertainty in the inferred total amount of compounds in V is, at most, proportional to the uncertainty of measured values and proportional to $e(z^T)$. Hence $e(z^T)$, which depends only on A, serves as an important measure of accuracy of the inferred results. If z^T ist the unit vector with the ith entry being 1, then $e(z^T)\epsilon$ bounds the deviation of the value of the ith variable, since $z^Tx = x_i$.

Here some discussion is in order. In the mentioned application the uncorrputed system $Ax = b$ always has a solution, for trivial reasons. System $Ax = b$ may lack a unique solution, but in our examples this is often the case only due to undistinguishable variables that always appear together. We can merge such variables (declare their sum a new variable). That is, subsets of variables may still have uniquely determined sums. Moreover, Definition 1 and Theorem 1 obviously do not become more complicated for general vectors z^T, therefore we consider also subsets of variables, even though we primarily want the values of single variables. The disturbed system $Ax' = b'$ usually becomes inconsistent, because some rows of A are linearly dependent, but the errors in b' are "uncorrelated".

We refer to the problem in Definition 1 as Bound Minimization. By a well-known standard trick, minimizing an ℓ_1-norm $\sum_i |y_i|$ under linear constraints can be written as an LP, hence we can compute $e(z^T)$ for any given z^T by an LP solver. (Introduce a new variable s_i for each $|y_i|$ and new constraints $s_i \geq y_i$ and $s_i \geq -y_i$, then minimize $\sum_i s_i$. Then any optimal solution satisfies $s_i = |y_i|$ for all i.) However in relevant special cases we can apply simpler combinatorial methods to Bound Minimization, as we will see.

For non-zero vectors z^T we always have $e(z^T) \geq 1$, and $e(z^T) = 1$ holds if and only if z^T is a row of A. We omit the proof of this simple observation.

3 Unit Vectors and Rows with Two Variables

In the following, all matrices A have only entries 0 and 1. In this section we consider the speical case of Bound Minimization when all rows of A have at most two entries 1. (Experimental results suggest that this case is of particular interest in our application, see the example later on.) We define a graph G such that A is the incidence matrix of G, that is, columns and rows of A correspond to vertices and edges, respectively, of G, and an entry 1 means that the vertex belongs to the edge. In particular, a row with exactly one entry 1 corresponds to a loop. Therefore, when z^T is a unit vector, we can obviosuly reformulate Bound Minimization as an edge labeling problem in graphs.

Edge Labeling: We are given a graph G, possibly with loops, and a distinguished vertex v. The problem is to label the edges (including the loops) of G with real numbers, thereby minimizing the sum of *absolute values* of all edge labels, under the following constraint: Let $S(u)$ denote the sum of labels of all edges incident to vertex u. Then $S(v) = 1$, and $S(u) = 0$ for all $u \neq v$.

If several edge labelings satisfy the above constraints, then any convex linear combination of them, i.e., linear combination with positive coefficients that sum up to 1, is also a feasible solution to Edge Labeling.

Assume that $Ax = b$ has a solution x. Also assume that G is connected, otherwise the following reasoning holds in every connected component. It is folklore in linear algebra that x is uniquely determined if G has a loop or an odd

cycle. Otherwise G is bipartite, and $Ax = b$ has a 1-dimensional solution space. In the following we extend this result to error propagation, in the sense that we characterize the optimal solutions to Bound Minimization. Here, a *tour* is a path in a graph, where edges may be traversed several times in any direction.

Theorem 2. *If graph G has neither loops nor odd cycles (bipartite graph), then* Edge Labeling *has no solution. Otherwise there exists a labeling. Furthermore, there exists an optimal labeling of one of the following two types, where edges not mentioned get label* 0*:*
(i) Take a shortest path from v to some loop, and assign labels $+1$ and -1 alternatingly to its edges, including this loop.
(ii) Take a shortest tour of odd length, and without loops, from v to itself, thereby adding alternatingly $+1/2$ and $-1/2$ to the labels of traversed edges.
Moreover, a tour of type (ii) consists of a path P (maybe empty), followed by an odd cycle C and the reverse of P.

Proof. For the moment we relax the constraint in Edge Labeling and call a labeling *valid* if $S(v) > 0$, and $S(u) = 0$ for all $u \neq v$. Denote by $s > 0$ the smallest absolute value in a given labeling. Any valid labeling can then be changed into a correct labeling by dividing all labels by $S(v)$. We shall see that successive subtraction of type (i) and (ii) labelings from a given labeling eventually produces a zero labeling, i.e., where all labels are zero.

So, consider a given optimal labeling. We start at v a tour T through, alternatingly, positively and negatively labeled edges. At the same time we subtract s from the absolute value of the label of every traversed edge. Observe that there cannot exist an even cycle where positive and negative labels alternate, since then subtraction would not alter any $S(u)$, contradicting optimality of the initial labeling. Hence a tour T with the following properties exists:

- T terminates at a loop or at v.
- In the former case, T has no other loops.
- in the latter case, T is free of loops and has odd length.
- Subtraction of labels (as described) leaves a new valid labeling.

Hence we find another tour in the graph equipped with a new valid labeling, and so on, until we get the zero labeling. In the event that v gets isolated, that is, all edges incident to v have labels reduced to 0, property $S(u) = 0$ for all $u \neq v$ implies that all labels in the graph are already 0 as well, since otherwise the initial labeling was not optimal. It follows that any given optimal labeling is a convex linear combination of labelings of type (i) or (ii). It also follows that loop-free bipartite graphs have no labeling at all.

In the set of valid labelings obtained in the above construction, every edge has a fixed sign: either positive or negative or zero. Moreover the sum of absolute values of labels is a linear function on the restricted set of labelings with those fixed signs. Since a linear function on a convex polytope achieves its optimum at some extremal point, it follows that some optimal labeling is of type (i) or (ii). Now, $s = 1$ in case (i) and $s = 1/2$ in case (ii) is enforced by the constraint

in the problem statement. Finally, optimality yields that T is a shortest tour of the respective type.

Consider type (ii). If a vertex w appears at least twice in T and the subtour between two copies of w has even length, we can delete this subtour from T, which contradicts optimality. If w appears at least three times in T, then some subtour between two of the copies has even length, which was already excluded. Thus w appears at most twice in T. Now let w be any vertex that actually appears twice. Such w exists, since at least v appears twice. Now we choose w such that all vertices on the subtour between the two copies of w are distinct; note that such w exists. Let P, C, Q denote the subtours from v to w, w to w, and w to v. Then C is an odd cycle. Since T has odd length, P and Q are both even or both odd. If P is no longer than Q, we can replace Q with the reverse of P without increasing the length of T, while keeping the total length odd. The case where P is longer than Q is symmetric. □

In the case of bipartite graphs, $e(z^T)$ for a unit vector z^T is undefined, however: Consider any two vertices u and v from different partite sets of G, and let z^T be the incidence vector of $\{u, v\}$. Then $e(z^T)$ exists and is the distance of u and v. More generally we have in any graph G, not necessarily bipartite:

Theorem 3. *Let u and v be any two vertices of G, and let z^T be the incidence vector of $\{u, v\}$. Then $e(z^T)$ is the minimum length of an odd path between u and v, and $e(z^T)$ is undefined if no such odd path exists.*

We omit the proof which is very similar to Theorem 2. Optimal tours in Theorems 2 and 3 can be found by adaptations of standard algorithms for shortest paths.

We conclude the section with a few remarks.

We can get upper bounds on errors also by elimination: In the following we use $e(x_i)$ as a shorthand for $e(z^T)$ where z^T is the indicator vector of x_i. As observed earlier, $e(x_i) = 1$ if x_i appears as the only variable in some row. By induction and the triangle inequality we obtain: If some row has the form $x_i + \sum_{k \in K} x_k$, where $\sum_{k \in K} e(x_k) \leq t$, then $e(x_i) \leq t + 1$. This can be extended straightforwardly to subsets of variables.

We emphasize that our results give worst-case upper bounds, while the actual error propagation should be much better. We illustrate this for case (i) of Theorem 2 (which is responsible for the error bounds of most variables in our simulation examples). The worst case appears only if the edges on the path are alternatingly labeled $+1$ and -1. To be specific, consider the following sequence of edge labels (beginning with the loop): $+1, -1, +1, -1, +1, -1 \ldots$ Then Gaussian elimination along this path yields the following error factors of variables: $+1, -2, +3, -4, +5, -6, \ldots$ (i.e., these are the factors in front of ϵ). Now the point is that, in real data, the errors in vector b which yield the edge labels will barely form such alternating paths. Assuming, for instance, that measurement errors are random and independent, the sequence of error factors will be a random walk that is unlikely to move very far away from 0.

4 Properties of Optimal Combinations of Rows

We come back to Bound Minimization in general, that is, the computation of $e(z^T) = \min\{||y^T||_1 : \ y^T A = z^T\}$for a given matrix A and vector z^T. In our sparse matrices we observe that optimal linear combinations are almost always built from a few rows, and often by just applying Theorem 2 (i). Hence optimal y are easier to obtain than by running an LP solver on relatively large matrices. The sparsity of optimal vectors y is partly explained by the fact that rows of A are sparse, and there always exists an optimal y where the set of rows of A with nonzero coefficients is linearly independent. (The latter claim follows by standard arguments from LP theory that we omit.)

On the other hand, the possible numerical values of $e(z^T)$ are dense, e.g., they can be arbitrarily close to 1. As an example that gives the idea, consider an $n \times n$ matrix with one row of only 1s, and $n-1$ other rows where x_1 appears together with any one of the variables $x_2, \ldots, x_n$. Then $e(1, 0, \ldots, 0) = 1 + \frac{2}{n-2}$. In general matrices A this means that linear combinations of rows that yield $e(z^T)$ (as in Definition 1) may involve rows with arbitrarily many 1 entries.

Therefore it would be nice to confirm optimality of a vector y obtained by a combinatorial method, without actually solving an LP. The natural idea to get an optimality certificate is to *dualize* the LP for Bound Minimization. Straightforward calculation and algebraic manipulation yields the following dual problem: $\max z^T r$, such that $-\bar{1} \leq Ar \leq \bar{1}$ (where entries of r may be negative). To confirm a value $e(z^T)$ in the case of unit vectors z^T, we only have to set the corresponding entry of r to the alleged value and choose other entries so as to keep all entries of Ar in the range from -1 to $+1$. Since y usually involves only a few rows, some sparse r is often quickly found in an ad-hoc way, without running an LP solver on the entire dual problem.

5 Approximate Solutions with Two Variables per Row

Theorem 1 and the subsequent discussion suggests a problem of independent interest. Given a $0,1$-matrix A and a vector b, we actually want to compute x such that ϵ is minimized in $|Ax - b| \leq \epsilon\bar{1}$.

If A has at most two entries 1 per row, this yields another graph labeling problem, on the incidence graph G of A:

Vertex Labeling: Given a graph with edges labeled by real numbers, label the vertices by real numbers such that the label of every edge differs from the sum of labels of its vertices by at most ϵ.

The problem can be transformed into an LP, but again we would like to have simpler combinatorial algorithms and structural insights.

The case of bipartite incidence graphs G is of special interest, since already the exact problem has no unique solution. In the following we give a combinatorial characterization of an optimal vertex labeling. Given a solution, i.e., some

vertex labeling, where the maximum deviation is ϵ, we call an edge high (low) if the sum of its vertex labels equals the edge label plus (minus) ϵ.

Theorem 4. *For* VERTEX LABELING *in a bipartite graph we have: If there exists an alternating cycle of high and low edges, then the vertex labeling is optimal. Otherwise there exists a vertex incident with some high edge but not incident with any low edge, or vice versa.*

Proof. Let C be an alternating cycle. In order to improve the labeling we must get rid of all high and low edges, in particular those in C. Let e be any high edge in C. We must decrease the label of some vertex v of e by some $\delta > 0$. But since v is also incident with a low edge, we must increase the label of its other vertex by more than δ, and so on. In this way we go round the cycle and eventually arrive at v whose label must be decreased, but now by more than δ, a contradiction.

By definition, there always exists some high or low edge. If every vertex incident with some high (low) edge is also incident with some low (high) edge, we can obviously form an alternating cycle. Now both assertions are proved. □

Theorem 4 suggests a simple algorithm for VERTEX LABELING in bipartite graphs: Take a vertex incident with some high (low) edge but no low (high) edge and decrease (increase) its label. This removes some high or low edge, or improves ϵ. This is repeated until an alternating cycle appears. Of course, worst-case time analysis would require some more work, along the lines of similar algorithms for flow problems.

6 Some Experimental Results and Conclusions

Remember from Theorem 1 that $e(z^T)$ of a unit vector z gives an error bound for the corresponding variable. To get an idea of the actual distribution of these errors we studied 100 matrices obtained from mixtures of 20 proteins randomly picked from Swissprot database, with trypsin digestion simulated in silico. Input matrices A are then constructed as explained in the Introduction. Instead of peptide identities (sequences) we distinghuished them only by their masses. Clearly, peptide identity information would have further improved the results.

We evaluated the $e(z^T)$ values for each variable by using both EDGE LABELING (Theorem 2) and the LP considering all rows of the matrix, and it seems that in almost all cases where EDGE LABELING manages to find a bound, these results strongly agree. In spite that for few variables $e(z^T)$ are undefined or bounds are scaringly high, the average values are quite reassuring. For the majority of variables $e(z^T)$ is still small, taking values at most 4, with an average below 3. The low $e(z^T)$ values persist also for larger mixtures; we tried some with 50 proteins in the mixture, and with further candidate proteins inserted to account for undetected peptide masses.

As a more detailed illustration we report one typical example. To simplify notation we write i for x_i, every row is written as the set of variables appearing

there, without parantheses, and separated by commas. We use symbol e to denote the error bound. Sets of variables with $e = 1$ are the given rows sorted by the number of variables. In particular, single variables with $e = 1$ are the proteins identified from unique peptides. Then we list the variables for every bound $e > 1$ that occured. The example contains 42 variables (candidate proteins). After merging of variables that appear only together, and therefore cannot have uniquely determined individual values, there remain 40 distinct variables.

e=1: 1, 2, 10, 14, 15, 16, 21, 22, 27, 28, 29, 32, 33, 34, 35, 38, 40, 41, 42,
1 21, 1 28, 2 15, 2 22, 2 24, 2 29, 3 21, 4 15, 5 28, 7 29, 8 33, 9 21, 9 27, 12 27, 12 28, 12 42, 13 17, 14 17, 14 22, 14 27, 15 17, 15 35, 16 30, 16 31, 20 28, 22 36, 23 29, 26 29, 27 29, 29 34, 29 40, 33 35, 37 41,
1 6 23, 1 11 37, 2 18 19, 2 28 41, 9 29 41, 10 29 41, 18 19 41, 23 24 41, 25 28 31, 27 33 40, 1 23 24 42, 23 24 35 41,
14 27 28 35 36, 15 23 25 26 42, 23 25 27 41 42,
20 21 24 25 27 37, 11 14 24 25 27 35 41,
1 2 10 12 14 21 23 24 27 29 33 34 35 41 42,
1 2 14 15 16 22 23 24 27 28 29 31 35 37 38 39 40 41 42

e=2: 3, 4, 5, 7, 8, 9, 12, 17, 18, 20, 23, 24, 26, 30, 31, 36, 37

e=3: 13; **e=3.67:** 11; **e=3.83:** 25; **e=4:** 6

The solution to this system turns out to be uniquely determined, subject to the few merged variables. (By elementary linear algebra, this fact depends only on A but not on b.) Remarkably, almost half of them have $e = 1$ and $e = 2$, respectively, and only a small rest have $e \leq 4$. We emphasize that many rows consist of one or two variables, and the optimal linear combinations almost exclusively use these graph edges and loops. The graph also exhibits some cycles, but only case (i) from Theorem 2 takes effect in this example, because loops are abound. Only a few optimal error factors are "hidden more deeply". For instance, $e \leq 4$ for variables 11 and 25 is easy to find without computer help, but the optimal result detected by GLPK is slightly better: $e = 3.67$ for variable 11 is a result of the following rows:
1, 21, 28, 35, 20 28, 37 41, 1 11 37, 20 21 24 25 27 37, 11 14 24 25 27 35 41,
with coefficients -2,+1,-1,-1,+1,-1,+2,-1,+1, all divided by 3.

We observe that the matrix structure is rich enough to give almost all variables uniquely determined values (where the only exceptions we found were sets of proteins with equal peptide mass spectrum), and at the same time their accumulated errors are small. Moreover, optimal combinations of measurements are mostly built from very sparse equations using a bit of elementary graph theory. (Thus the restriction to rows with at most two 1s is not as narrow as it may seem.) In some cases the LP gave somewhat better combinations, but this degree of accuracy does not seem to be very important, as the $e(z^T)$ merely have the role of pessimistic worst-case bounds; see the earlier remark.

We conclude that quantitative mixture reconstruction is possible when sufficiently accurate measures are available. At least this does not fail already for intrinsic mathematical reasons, i.e., by the structure of the incidence matrices. Although we could test, at this stage, only simulated (i.e., random) mixtures, we expect similar matrix structures also for real mixtures. It must be admitted that our simulations did not take undetectable peptides into account, i.e., the set of available rows in the matrices may be more limited. However we also notice that most unit vectors have several alternative representations as "small" linear combinations of rows, and it suffices that some of these row sets is present. It was also pointed out in [6] that improved accuracy of the measurement technology will increase the power of mathematical inference methods. Another type of errors not considered here is that some proteins could very well be in the mixture but are ignored because their masses are not listed among the observed ones, although some of the masses they share with other proteins are detected. We are planning to address this issue in the future by extending the selection of columns included in the input matrix A. However, current quantification procedures consider only unique peptides, and thus the first results for shared peptides are a step forward.

Other directions of further work include: to examine larger simulated mixtures and real mixtures, to derive both simple upper bounds and combinatorial optimality criteria for Bound Minimization also when rows have some more 1s entries (however the resulting hypergraph problems could be intrinsically more complicated), to give a theoretical explanation of the observed limited error propagation by "helpful" properties of the matrices, and to study probabilistic error models. In our study we adopted a simple error model to start with, assuming a uniform error bound ϵ in b. Due to practical considerations we may want to allow for different errors of the b_i, a "vector-ϵ" so to speak, which may be known or unknown in advance. On the other hand, our exact systems $Ax = b$ typically have uniquely solutions (after removal of duplicate columns) and are even largely overdetermined. These two points give rise to the following type of problem:

Given a matrix A with full column rank, and a corrupted vector b' coming from an unknown $b = Ax$, can we efficiently compute x' such that in Ax' most entries are close to the given ones in b', with only few outliers? In particular, can we recognize which b_i had small and large errors in a particular instance? What global error assumptions on b are sufficient to enable such an approximate reconstruction of x? It is clear that some error assumptions must be made. Also, this is not a single clear-cut problem but rather a research direction. The work in [6] belongs to this direction and proposes some error norm, but can also think of alternative ones. For an $m \times n$ matrix A this problem amounts to a geometric one: reporting and clustering the intersection points of subsets of n linearly independent hyperplanes in an arrangement of $m > n$ hyperplanes in n dimensions, perhaps under additional assumptions how the arrangement was obtained. The problem can also be formulated as finding maximal feasible

subsystems of linear equations. Negative complexity results as in [7, 10] do not rule out the possibility of efficient parameterizations.

Acknowledgments

This work has been supported by the Swedish Research Council (Vetenskapsrådet), grant no. 2010-4661, "Generalized and fast search strategies for parameterized problems". Early stages of the second author's work have also been supported by Devdatt Dubhashi through a Chalmers Bioscience Initiative grant. The authors thank Azam Sheikh Muhammad for some discussions and for valuable technical help in preparing data and scripts for GLPK.

References

1. Arioli, M., Demmel, J.W., Duff, I.S.: Solving Sparse Linear Systems with Sparse Backward Error. SIAM J. Matrix Analysis and Appl. 10, 165–190 (1989)
2. Bantscheff, M., Schirle, M., Sweetman, G., Rick, J., Kuster, B.: Quantitative Mass Spectrometry in Proteomics: A Critical Review. Anal. Bioanal. Chem. (2007) 389, 1017–1031 (2007)
3. Boehm, A.M., Pütz, S., Altenhöfer, D., Sickmann, A., Falk, M.: Precise Protein Quantification Based on Peptide Quantification Using iTRAQ. BMC Bioinformatics 8, 214 (2007)
4. Chandrasekaran, S., Ipsen, I.C.F. On the Sensitivity of Solution Components in Linear Systems of Equations. SIAM J. Matrix Analysis and Appl. 16, 93–112 (1995)
5. Damaschke, P.: Sparse Solutions of Sparse Linear Systems: Fixed-Parameter Tractability and an Application of Complex Group Testing. In: Marx, D., Rossmanith, P. (eds.) IPEC 2011. LNCS, vol. 7112, pp. 94–105, Springer, Heidelberg (2011)
6. Dost, B. Bafna, V., Bandeira, N., Li, X. Shen, Z. Briggs, S.: Shared Peptides in Mass Spectrometry Based Protein Quantification. In: Bazoglou, S. (ed.) RECOMB 2009. LNCS, vol. 5541, pp. 356–371, Springer, Heidelberg (2009)
7. Feige, U., Reichman, D.: On the Hardness of Approximating Max-Satisfy. Inf. Proc. Lett. 97, 31–35 (2006)
8. Fritzilas, E., Milanic, M., Rahmann, S., Rios-Solis, Y.A.: Structural Identifiability in Low-Rank Matrix Factorization. Algorithmica 53, 313–332 (2010)
9. Gerber, S.A., Rush, J., Stemman, O., Kirschner, M.W., Gygi, S.P.: Absolute Quantification of Proteins and Phosphoproteins from Cell Lysates by Tandem MS. Proc. Nat. Academy of Sc. USA 100, 6940–6945 (2003)
10. Giannopoulos, P., Knauer, C., Rote, G.: The Parameterized Complexity of Some Geometric Problems in Unbounded Dimension. In: Chen, J., Fomin, F. (eds.) IWPEC 2009. LNCS, vol. 5917, pp. 198–209, Springer, Heidelberg (2009)
11. Nesvizhskii, A.I., Aebersold, R.: Interpretation of Shotgun Proteomic Data: The Protein Inference Problem. Mol. Cellular Proteomics 4, 1419–1440 (2005)
12. Yang, X., Dai, H., He, Q.: Condition Numbers and Backward Perturbation Bound for Linear Matrix Equations. Num. Lin. Algebra with Appl. 18, 155–165 (2011)
13. Zhang, B., Chambers, M.C., Tabb, D.L.: Proteomic Parsimony through Bipartite Graph Analysis Improves Accuracy and Transparency. J. Proteome Res. 6, 3549–3557 (2007)