Solving Multiple Inference by Minimizing Expected Loss

Cong Chen CONG.CHEN@QC.CUNY.EDU
Jiaqi Yang JYANG2@GRADCENTER.CUNY.EDU
Chao Chen CHAO.CHEN.CCHEN@GMAIL.COM
Changhe Yuan CHANGHE.YUAN@QC.CUNY.EDU
The City University of New York and Stony Brook University

Abstract

Multiple Inference is the problem of finding multiple top solutions for an inference problem in a graphical model. It has been shown that it is beneficial for the top solutions to be diverse. However, existing methods, such as diverse M-Best and M-Modes, often rely on a hyper parameter in enforcing diversity. The optimal values of such parameters usually depend on the probability landscape of the graphical model and thus have to be tuned case by case via cross validation. This is not a desirable property. In this paper, we introduce a parameter-free method that directly minimizes the expected loss of each solution in finding multiple top solutions that have high oracle accuracy, and are automatically diverse. Empirical evaluations show that our method often have better performance than other competing methods.

Keywords: Graphical Model; Multiple Inference; Oracle Accuracy; Expected Loss.

1. Introduction

For inference problems, much effort has been directed at algorithms for obtaining one single optimal prediction. In reality, however, the data are sometimes corrupted or incomplete, which makes it necessary to increase the confidence in the answer via finding several best solutions where multiple hypotheses are preferred for advanced reasoning. Multiple inference has shown impressive results in a number of computer vision (Batra et al., 2012; Yadollahpour et al., 2013; Kirillov et al., 2015) and computational biology (Fromer and Yanover, 2009), and machine translation (Gimpel et al., 2013).

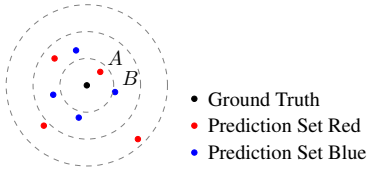


Figure 1: Illustration of the concept of oracle accuracy. The central black node represents the ground truth. The red and blue nodes represent two sets of predictions. The best prediction (A) of red set is closer to the ground truth than the best prediction (B) of the blue set, in spite that the blue set is generally all closer to the ground truth. Therefore, the oracle accuracy of set red is higher than the oracle accuracy of set blue.

It's hard to tell which prediction is better than others from multiple results, thus, ideally after all, we expect one of the given best solution candidates would be chosen as the final answer, i.e. the chosen solution should have a very high accuracy. *Oracle accuracy* is used as evaluation criteria which is defined as the highest accuracy of one of the predictions compared to the ground truth. See Figure 1. The problem becomes finding a set of solutions which has highest oracle accuracy.

In order to solve the problem, people have proposed different methods targeting better results. M-Best inference (Dechter et al., 2012) (Figure 2(a)) obtains the M most probable predictions. Diverse inference tries to find a set of solutions with both high probability and high diversity. Existing methods, such as diverse M-Best (Batra et al., 2012) (Figure 2(b)), which iteratively finds M dis-

similar high probable solutions, and M-Modes (Chen et al., 2013) (Figure 2(c)), which computes the top M local optima, rely on a hyper parameter in enforcing diversity.

Current approaches for solving multiple inference are all derivatives of finding the posterior mode(s), aka the MAP estimates. These methods climb the probability distribution landscape: Routed from the top (MAP), then either choose top one, or jump to farther ones preventing similarity. The MAP estimation targets to find high probable point(s) without taking the volume of the landscape into account.

However, it is important to point out that MAP estimation has drawbacks. The mode(s) are usually quite untypical of the distribution (Murphy, 2012). When the loss function is beyond 0-1 loss, such as Hamming, besides when the problem is multiple predictions, choosing the mode(s) is often a very poor choice. Different from MAP estimation, Bayesian methods are characterized by the use of distributions to summarize data and draw inferences. Why don't we directly optimize oracle accuracy by more general Bayesian methods without resorting the MAP estimation? This belief will provide motivation for a more fundamental Bayesian approach instead.

In contrast to current MAP-based inference, our new objective remodels optimizing oracle accuracy, and directly minimizes expected loss in finding high-accuracy multiple solutions. Our new method Min-Loss M-Best (Figure 2(d)) aims to jointly find M solutions, which at least one of them has the lowest expected loss. We will first discuss some theoretical understandings of Min-Loss M-Best properties, then develop practical solutions for solving it. In particular, due to the high computational complexity of solving Min-Loss M-Best, we use top M-Best solutions to approximately simulate the whole distribution, then search the best possible choices from these M solutions. We implement and test our proposed method with current MAP-based methods, and empirical evaluations show that our proposed method has better oracle accuracy than other competing methods.

2. Background

We begin by providing background on multiple inference on probabilistic graphical models.

2.1 Probabilistic Graphical Models

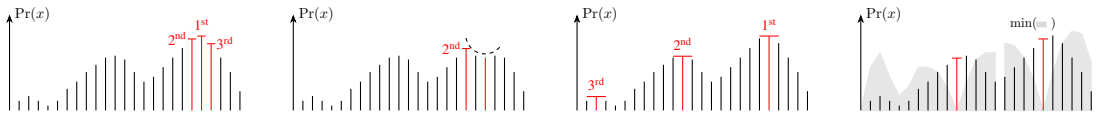
A *probabilistic graphical model* (PGM) is a collection of local functions over subsets of variables that conveys probabilistic information. The structure of a PGM can be visualized as a graph. The graph captures independence inherent in the model that can be useful for interpreting the modeled data and be exploited by inference tasks (Wainwright and Jordan, 2008; Flerova et al., 2016).

A PGM consists of a finite set of discrete random variables, V, a set of non-negative real-valued discrete local functions over scopes variables, F, and a combination operator, $\odot$. The model represents a global function, which is a combination of all the local functions.

Random variables correspond to vertices in the graph. The terms of variable, node, and vertex are used interchangeably. A discrete value assigned to a variable is called a label. A label assignment for all variables is called a labeling or configuration. We use lowercase letters, x, y, to represent a labeling. Local functions are functions over a set of variables and can also be called potentials.

When a graphical model's operator $\odot = \prod$ and local functions $f(v) = \Pr\big(\, v \mid \mathrm{parent}(v) \,\big)$, we have a *Bayesian network* (Pearl, 1988). When a graphical model's operator $\odot = \sum$ and $f_c(x_c) = -\log\big(\, \psi(x_c) \,\big)$, where c are the set of all maximal cliques, we have a *Markov random field*.

We denote $f(x) = -\log\big(\prod_{c \in \mathcal{C}} \psi(x_c)\big)$ as the energy of labeling x, where $\psi(x_c)$ is the potential function over a maximal clique x_c. The energy is proportional to the negative log probability.



(a) M-Best (b) Diverse M-Best (c) M-Modes (d) Min-Loss M-Best

Figure 2: Illustration of four multiple inference methods. Each vertical bar corresponds to a labeling while the red bars represent the predictions of each method. The height of the bar corresponds to the labeling's probability. (a) M-Best: it uses M most probable solutions as predictions; (b) Diverse M-Best: it shows the second solution is from the diversity suppression of the first solution; (c) M-Modes: it represents the solutions are top local optimal solutions where neighborhood size $\delta = 1$; (d) Min-Loss M-Best: it shows the two solutions are selected when the whole expected loss (the gray shades) are minimum.

Maximum a posteriori (MAP) probabilities, also known as the *most probable explanation* (MPE), computes the highest probability labeling over the probability distribution. In the context of inference applications, the labeling solution is also called the prediction. Let $\mathcal{Y}$ be the set of all the labelings $y_1, y_2, \ldots$, of a model occurring with probabilities $\Pr(y_1), \Pr(y_2), \ldots$, respectively. Its solution is the global maximum labeling y^* of the probability landscape. This task is also the same as finding a labeling y^* which minimizes the corresponding energies $\mathrm{E}(\cdot)$:

$$y^* = \arg\max_{y \in \mathcal{Y}} \Pr(y) = \arg\min_{y} \mathrm{E}(y) \tag{1}$$

2.2 Multiple Inference

In reality, however, the data are sometimes corrupted or incomplete, which makes obtaining a single optimal solution questionable. It is necessary to increase the confidence in the answer via finding several best solutions. Then we can ask an expert to choose a final solution (Flerova et al., 2016), rank and combine a very large pool (Li et al., 2010), or even further improve the solutions in a human-in-the-loop environment.

M-Best is to obtain the M most probable labelings over the probability distribution. This problem has been well studied from these approaches: 1) k shortest path like methods by detouring the current solutions (Lawler, 1972; Nilsson, 1998), 2) M-Best extension of LP-relaxation (Fromer and Globerson, 2009), and 3) Heuristic search such as M-A* and M-BB (Flerova et al., 2016). However, these top best solutions tend to be very similar to the MAP solution or to each other, thus lacking diversity. See Figure 2(a).

Diverse Multiple Inference provides a principled way to trade off dissimilarity versus probability. It highlights the concept *diversity* in the top solution set. The goal is to find a set of high-quality solutions that are also qualitatively different from each other. In order to make sure solutions are qualitatively different, we need a distance measure between solutions.

A dissimilarity function $\Delta(\,\cdot\,)$ is used to define the distance between several labelings, i.e. $\Delta(\,\cdot\,)$ takes a large value if labelings are different, and a small value otherwise. This distance measure can be classified into nodewise and pairwise distances (Kirillov et al., 2015). Hamming distance is a special case of both nodewise and pairwise distances, besides an indicator function for each disagreed variable values. Without loss of generality, in this paper, we assume using Hamming distance.

Diverse M-Best (Batra et al., 2012) algorithm starts with the MAP as the first solution, then iteratively and greedily finds next solutions via a regularization of a diversity penalty term. This penalty term makes the next solution be dissimilar by a certain margin from the solutions collected

so far:

$$y^* = y_1 = \arg\min_y \Big[\mathrm{E}(y) \Big] \tag{2}$$

$$y_m = \arg\min_y \Big[\mathrm{E}(y) - \lambda \sum_{i=1}^{m-1} \Delta(y, y_i) \Big] \tag{3}$$

Let y_m denote the m^{th}-best solution, thus y_1 is the MAP, y_2 is the 2^{nd}-best, and so on. See Figure 2(b). The λ coefficient measures how strong the penalty term is. An appropriate value of λ is problem-dependent, and it could be tuned by cross-validation, i.e., we learn the appropriate degree of diversity by tuning λ on a validation data set.

M-Modes (Chen et al., 2013) method computes the top M locally optimal configurations, each of which has higher quality than all other solutions within a given scalar distance δ. These locally optimal solutions, called *modes*, capture the topographical features of the probabilistic landscape of the given graphical model, and are also highly possible and are naturally diverse. See Figure 2(c).

Given a non-negative integer δ, δ-neighborhood $\mathcal{N}_\delta(y)$ is defined as $\mathcal{N}_\delta(y) = \{\, y' \mid \Delta(y', y) \leqslant \delta \,\}$. So, a labeling y is a δ-mode as y has highest probability (lowest energy) in its δ-neighborhood. Therefore, M-Modes is an algorithm to compute the top M best modes. This definition ensures the modes are diverse; any two modes are at least δ away.

Oracle Accuracy is often used as the evaluation criterion in multiple prediction, i.e., the highest accuracy of one of the M predictions compared to the ground truth, y_{gt}, as commonly done in the multiple prediction literature (Batra et al., 2012; Gimpel et al., 2013; Chen et al., 2018).

In the context of multiple inference tasks, when the model has already correctly depicted the real distribution, a ground truth labeling can be regarded as a sample drawn from the model distribution, i.e., $y_{\text{gt}} \sim \mathcal{Y}$. Oracle accuracy doesn't care about whether the remaining solutions are poor quality or not, but one solution with the highest accuracy is accepted. The oracle accuracy of a set of M labeling, $\{y\}_M$, by Hamming distance, can be calculated as:

$$\mathsf{OrcAcc}\big(\{y\}_M\big) = \frac{|y| - \min\big(\Delta(y_1, y_{\text{gt}}), \ldots, \Delta(y_M, y_{\text{gt}})\big)}{|y|} \tag{4}$$

Here, the $|y|$ represents the variable size of y. Often, we also use *error rate* instead of accuracy which is $\mathsf{ErrRate}(y) = 100\% - \mathsf{OrcAcc}(y)$

3. Min-Loss Multiple Inference

3.1 Motivation

There is no solid proof that finding these posterior modes can always luckily obtain good results by applying some diversity constraints over with high probability points. Both Diverse M-Best and M-Modes have critical drawbacks that they require tuning diversity parameters, namely either λ or δ, by cross-validation. As λ is a continuous numeric, we cannot easily find an optimal value for all problems in Diverse M-Best. The δ in M-Modes is an integer value, but its optimum value is varying for different cases. If λ or δ is set too low, the next solution may still be trapped at same peak; if too high, many good solutions will be ignored. This problem is related with the topology of the distribution landscape. Consequently, these methods do not consider the landscape as a whole picture.

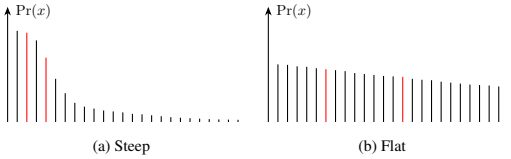


Figure 3: Suggested multiple solutions (red bars) for steep and flat probability distribution landscapes.

This raises a concern about choosing solutions on different distribution landscapes. See Figure 3. When the landscape is steep, where some of the solutions have very high probabilities, it is more profitable to select from the peak and sacrifice diversity (Figure 3(a)); when the landscape is flat, where all of the labelings have similar probabilities, more diverse choices would give higher chance to be close to the ground truth (Figure 3(b)). In addition, for M-Modes solutions, if there is only one big hill, e.g., many models assuming Gaussian distribution, the MAP is its only mode solution. We cannot generate more candidates.

Therefore, we should tune the hyper parameter automatically. We agree with the opinion that diversity should not explicitly enforced ad hoc, and should be an emerging property (Dey et al., 2015) reflecting different distribution landscapes. Bayesian methods are characterized by the use of distributions to summarize data and draw inferences. Directly and jointly optimizing oracle accuracy for M solutions by more general Bayesian methods should be much better than diversifying the modes. We can create a more fundamental parameter-free approach.

Table 1: An Example

LBL	PR
000	.15
001	.14
010	.14
011	.14
100	.14
101	.14
110	.14
111	.01

(a)

2-SOLN	S100	S1,000	S10,000		S∞
001, **110**	.73	.716	.7201		.72
010, **101**	.77	.714	.7124		.72
011, **100**	.75	.732	.7173		.72
000, **111**	.76	.837	.8371	$\rightarrow$	.84
000, **001**	.94	.876	.8617		.86
000, **010**	.87	.855	.8613		.86
000, **100**	.91	.873	.8544		.86
...		...			...

(b)

An Example: We use an example to motivate our approach. Let us look at a toy problem over three variables in Table 1. Each variable has two labels: **0** and **1**; there are totally $2^3 = 8$ different labelings (col. LBL in (a)). The probabilities of all labelings (col. PR in (a)) are as shown and sum to 1.0. Based on the probability distribution in (a), we generated 3 random data sets which have 100, 1,000, and 10,000 data points (samples) as the ground truth. As a multiple inference problem, we exhaustively enumerate all the possible 2-solution labelings as our results; there are totally $\binom{8}{2} = 28$ different choices (pairs). For each choice, we assumed it as the true prediction and calculated average error rates on each data set (listed at col. S100, S1,000, and S10,000 in (b)). The evaluation we use is oracle accuracy on Hamming distance. For simplicity, we just list top 7 best pairs with lowest average error rates. As the number of samples increased, the rates seemed to converge to a limit (col. S∞ in (b)).

Surprisingly, the lowest error rate pairs do not include MAP labeling **000** with highest probability of 0.15. We can explain it as follows: It would have high accuracy but be lack of diversity if we chose **000** and any other 0.14 labelings, while it would have enough diversity but be lack of accuracy if we chose **000** and very low probability **111** (0.01). So that the best three pairs of choices do not include MAP labeling **000**.

From this example, we notice the MAP solution could not always be included in the optimal results. A good prediction should consider both its distance to other labelings and their probabilities, as the ground truth is drawn from distribution. In many machine learning evaluation tasks, a predicted solution which is different from the correct answer would not be forfeited all the points; using Hamming distance is one example. Therefore, using most probable solutions might not be a good choice because it does not consider distances between solutions. Using MAP does not necessarily

lead to high prediction accuracy. So, how can we quantitatively expect any prediction's distance from the ground truth? These observations motivate us to propose the concept of the expected loss of a labeling.

3.2 Expected Loss

Given any target labeling y for prediction, the *expected loss* or *loss* of y is the accumulation of the distance of each labeling y_{gt} from y weighted by probability of y_{gt}, i.e., the dot product of distances and probabilities:

$$\mathsf{loss}(y) = \mathop{\mathbb{E}}_{y_{\mathrm{gt}} \sim \mathcal{Y}} \left[\, \Delta(y, y_{\mathrm{gt}}) \, \right] = \sum_{y' \in \mathcal{Y}} \Pr(y') \cdot \Delta(y, y') \tag{5}$$

So that, if a generated data set has an enough large number of samples, the labelings' expected error rate (distance) should converge to this expected loss (See Table 1 col. S∞).

Based on the concept of expected loss, we define another useful optimization inference task called *Min-Loss MAP*:

Problem 1 (Min-Loss MAP)**.** *Find a labeling of a model which has lowest expected loss:*

$$y^* = \arg\min_{y \in \mathcal{Y}} \mathsf{loss}(y) \tag{6}$$

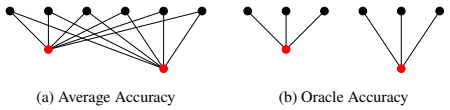
The solution of Min-Loss MAP can be quite different from MAP. MAP can be considered as a special case of Min-Loss MAP in which $\Delta(y, y_{\mathrm{gt}}) = [\![\, y \neq y_{\mathrm{gt}} \,]\!]$.

Common distance measures are nodewise distance, thus we can utilize this property to efficiently minimize each variable's distance. Then summing over these choices are naturally the global minimum.

Hamming distance is a nodewise distance, and it counts distances of different variables independently. Hence finding a minimal distance label of a variable (i.e. summation of other labels has minimum probability) is the same as a maximal marginal (i.e. current label has maximum probability). We can accordingly conclude such a theorem that max marginals can exactly solve the Min-Loss MAP problem using Hamming distance.

Theorem 1. *Max marginals $\Leftrightarrow$ Hamming Min-Loss MAP.*

Despite max marginals have high accuracy is a known observation in the literature, we should notice that this split rule can only be applied for Hamming loss.



(a) Average Accuracy (b) Oracle Accuracy

Figure 4: Diagrams shows the different measurements between (a) Average Accuracy and (b) Oracle Accuracy for $M = 2$. The red nodes represent the predictions and the black circles represent other labelings. The lines between red and black nodes represent the ways for distance measurement. For average accuracy, the loss takes all the labelings into account, while for oracle accuracy, it only measures the distances of closer predictions.

3.3 Min-Loss M-Best

Similar to M-Best multiple inference problems, we can also extend our Min-Loss MAP task to *Min-Loss M-Best*, which computes the top M solutions with lowest expected loss. But it is not the correct way for oracle accuracy when $M > 1$. Oracle accuracy consider the highest prediction as the distance measurement; while naive Min-Loss M-Best consider each prediction to the whole distribution, i.e. average accuracy. See Figure 4 for a better understanding of the difference. Therefore, optimizing oracle accuracy for Min-Loss M-Best problem is totally another story.

4. Min-Loss M-Best for Oracle Accuracy

Formally, optimizing oracle accuracy for M-solution problem by minimizing expected loss can be described as:

$$\{y\}^*_M = \arg\max_{\{y\}_M \in \mathcal{Y}} \mathsf{OrcAcc}\big(\{y\}_M\big) = \arg\min_{\{y\}_M \in \mathcal{Y}} \mathsf{loss}(\{y\}_M) \tag{7}$$

$$= \arg\min_{\{y\}_M \in \mathcal{Y}} \mathop{\mathbb{E}}_{y_{\text{gt}} \sim \mathcal{Y}} \Big[\min\big(\Delta(y_1, y_{\text{gt}}), \ldots, \Delta(y_M, y_{\text{gt}})\big) \Big] \tag{8}$$

$$= \arg\min_{\{y\}_M \in \mathcal{Y}} \sum_{y' \in \mathcal{Y}} \Pr(y') \Big[\min\big(\Delta(y_1, y'), \ldots, \Delta(y_M, y')\big) \Big] \tag{9}$$

This problem aims to jointly find M solutions, $\{y\}_M$, which at least one of them has lowest expected loss.

As the whole labeling distribution $\mathcal{Y}$ is exponentially large, we cannot find an exact algorithm to solve the Min-Loss MAP problem in a large model. However, it is possible to solve it approximately using M-Best solutions, i.e., let $\{y\}_{m_{\text{(Best)}}} \sim \mathcal{Y}$.

Once we have generated $m_{\text{(Best)}}$ solutions, we can form a pairwise loss table where each entry is a term that is the row's probability multiplies the distances of two solutions. For conciseness, we simplified the terms: $\Pr(y_m)$ as p_m, and $\Delta(y_m, y_n)$ as $\Delta_{m,n}$. This is a pre-processing step. See Figure 5. Note that the diagonal entries are zero as $\Delta_{m,m} = 0$.

	y_1	y_2	$\cdots$	y_{m-1}	$y_{m_{\text{(Best)}}}$
y_1	0	$p_1\Delta_{12}$	$\cdots$	$p_1\Delta_{1,m-1}$	$p_1\Delta_{1,m}$
y_2	$p_2\Delta_{21}$	0	$\cdots$	$p_2\Delta_{2,m-1}$	$p_2\Delta_{2,m}$
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
y_{m-1}	$p_{m-1}\Delta_{m-1,1}$	$p_{m-1}\Delta_{m-1,2}$	$\cdots$	0	$p_{m-1}\Delta_{m-1,m}$
$y_{m_{\text{(Best)}}}$	$p_m\Delta_{m,1}$	$p_m\Delta_{m,2}$	$\cdots$	$p_m\Delta_{m,m-1}$	0

Figure 5: Pairwise Loss Table from M-Best. The red shaded two columns are two chosen labelings (y_2 and y_{m-1}). The red circled terms are weighed distances to each labelings (rows). The target optimizing problem is to minimize the summation of these terms.

We can formulate finding an oracle accuracy M solutions as: Minimize the summation of all the entries such that 1) only one entry for each row can be selected and 2) the number of correlated columns should at most M. For no ambiguity, we use m for the input table size (from M-Best), and M for the number of solutions of oracle accuracy. We use *at most* since that it is acceptable to use fewer solutions to get the same results.

After compiling the table, we can formally define the optimization problem as follows:

$$\min \sum_{i=1}^{m}\sum_{j=1}^{m} \mu_{ij} \cdot (p_i \Delta_{ij}) \tag{10}$$

$$\text{s.t.} \quad \mu_{ij} \in \{0, 1\}, \quad \forall i, j \tag{11}$$

$$\sum_{j=1}^{m} \mu_{ij} = 1, \quad \forall i \tag{12}$$

$$\sum_{j=1}^{m}\prod_{i=1}^{m}(1 - \mu_{ij}) \geqslant m - M, \quad \forall i, j \tag{13}$$

Our target is to optimize over the indicator variables μ, with the constraints that force each row to choose only one entry.

Algorithm 1 Finding Min-Loss M-Best Solutions

Require: M-Best solutions: $\{y\}_m$; M
Ensure: Optimal M predictions: $\{y\}^*_M$
  function MIN-LOSS M-BEST($\{y\}_m$, M)
    $\{y\}_m$.sort(on = Pr)
    $loss_{\min} \leftarrow \infty$
    $\mathcal{Y}_M \leftarrow$ Combination($\{y\}_m, M$)
      or Sample($\{y\}_m, M$)
    for all $\{y\}_M \in \mathcal{Y}_M$ **do**
      $loss \leftarrow 0$
      for all $y_m \in \{y\}_m$ **do**
        $loss \leftarrow loss + p_m \cdot \min(\Delta_{m,M^1}, \ldots, \Delta_{m,M})$
        if $loss \geqslant loss_{\min}$ **then break**
      if $loss < loss_{\min}$ **then**
        $loss_{\min} = loss, \{y\}^*_M = \{y\}_M$
    return $\{y\}^*_M$

The product of the last constraint simulates an OR constraint, indicating the number of involving columns (labelings chosen) should be fewer than M. See Figure 5's red shades. Because the last constraint is not linear, we cannot make it as an integer linear programming problem.

Searching Optimal Min-Loss M-Best

We propose a search method: For small M, we can exhaustively search the possible choices y_M from top m solutions. And for each choice, we compute its oracle accuracy with respect to the m solutions. We report the choices with minimum expected loss, y^*_M. When we accumulate the distance, we prefer to expand labelings which have high probabilities. At the same time, current best solution is used as an upper bound to prune rest computation which has larger value than the bound. See Algorithm 1. This method has complexity $\mathcal{O}(\binom{m}{M} vmM)$.

When M is large, we would like to randomly sample M solutions out of m instead of exhaustively enumerating all the possible choices. In addition, we also use current best result to do the pruning, then run the method for certain amount of time and report the current best choices we found.

One interesting question is whether there are some heuristic functions which can help us speed up the search. Since getting more labelings will obtain lower distances, searching optimal Min-loss M-Best is not easy to be formulated as shortest path problems. One tentative way is to formulate the problem from m choosing M to m not choosing $(m - M)$, but it also raised another problem that the search space is even larger (because $m \gg M$). Finding better heuristic search algorithms to solve this problem is interesting but left as future work.

5. Experiments

We implement and test our proposed methods (Min-Loss M-Best), including exact and approximate algorithms for exhaustive and random search algorithms for oracle accuracy. We use M-Best, Di-

verse M-Best, and M-Modes as our baseline. We try to find general parameters for Diverse M-Best and M-Modes with best performances. We implement M-Best from Nilsson (1998)'s partition candidates method and Flerova et al. (2016)'s heuristic search method, Diverse M-Best from Batra et al. (2012), and M-Modes from Chen et al. (2018)'s heuristic search using tree decomposition.

Without loss of generality, we selected several discrete benchmark models from *bnlearn Bayesian Network Repository* [1], including two small networks, Asia (Lauritzen and Spiegelhalter, 1988) and Sachs (Sachs et al., 2005), two medium networks, Child (Spiegelhalter et al., 1993) and Alarm (Beinlich et al., 1989), and two large networks[2], Hepar2 (Onisko, 2003) and Win95pts. Our goal is to test the collective capability of predicting a set of top labelings ($M = 1, 3, 5, 7, 9$) with more oracle accuracy than M-Best, Diverse M-Best, or M-Modes. Hamming distance is used as distance measure. There is no exact methods for oracle accuracy except $M = 1$. We generated 1,000,000 random samples, as the ground truth, from each given network to prevent any bias of the data sets. Next, we test each methods' error rates (divided by number of variables; we omit the percent, %, the lower the better) over oracle accuracy on different M.

Our experiments were performed on an IBM System with 32 core 2.67GHz Intel Xeon Processors and 512G RAM. The program was written in C++ using the GNU compiler G++ on a Linux system. We just use one core for each program. We use functions from an aforementioned R package, *bnlearn* [3], for simulating random data from given Bayesian networks. We also use functions from *SMILE* [4] to construct the junction trees from the Bayesian networks. Last, accuracy evaluation on each methods' generated prediction sets is written in a MATLAB script.

5.1 Oracle Accuracy Performance

We list the results for all the networks, including two best performance parameters for Diverse M-Best and M-Modes. The results are shown in Table 2.

The number marked after the network name is its variables size. Two parameters for Diverse M-Best are listed, $\lambda = 0.1$ and 0.3, for either of them works well on these 6 models. Similar, we list two parameters for M-Modes, $\delta = 1$ and 3. Some of the datesets like Asia, Sachs, and Hepar2, do not have $\delta = 3$ modes. For our Min-Loss M-Best method, we use top 1000 M-Best solutions to compute. Asia is an exception because it is a small network that there are only 256 labelings, so we use them all. When M is small, like $M = 1, 3$, we use exact method, while we use random sample method instead when M is large, like $M = 5, 7, 9$. The program runs for one hour then report best found solutions so far. Exception for Asia since we only use 256 labelings, we can exactly compute $M = 5$.

Generally, Min-Loss M-Best is competitive in all the methods. For small M, Min-Loss M-Best can be exactly solved so that it has very good performance comparing to other methods. But when M is large, random methods sometimes weak inferior than some other methods.

Diverse M-Best has better performance on small networks setting $\lambda = 0.3$, and worse performance setting $\lambda = 0.1$. For example, Diverse M-Best has a very good performance where $\lambda = 0.3$ on Asia. But when it comes to large models, a small $\lambda = 0.1$ always works better than $\lambda = 0.3$.

1. www.bnlearn.com/bnrepository/
2. With respect to high complexity of M-Modes and Min-Loss M-Best, tens of variables' networks can be called "large". At bnlearn repository, these models also belongs to Large Networks.
3. www.bnlearn.com/
4. www.bayesfusion.com/

Table 2: Error rates (%) of oracle accuracy on benchmark models

$M=$	1	3	5	7	9
Asia (8)					
M-Best	20.46	6.61	4.40	3.54	1.55
Diverse M-Best ($\lambda=0.1$)	20.46	8.70	8.70	4.74	3.27
($\lambda=0.3$)	20.46	6.74	3.27	2.37	2.35
M-Modes ($\delta=1$)	20.46	6.74	6.44	–	–
Min-Loss M-Best (256)	20.45	6.61	3.27	3.21	2.86
Sachs (11)					
M-Best	37.81	31.08	27.87	26.32	25.12
Diverse M-Best ($\lambda=0.1$)	37.81	30.63	28.07	22.49	21.37
($\lambda=0.3$)	37.81	30.77	24.24	22.54	21.81
M-Modes ($\delta=1$)	37.81	31.08	24.38	23.33	21.99
Min-Loss M-Best (1000)	37.81	27.67	24.07	22.27	21.42
Child (20)					
M-Best	43.44	28.34	27.61	26.31	25.72
Diverse M-Best ($\lambda=0.1$)	43.44	28.34	27.52	25.55	24.60
($\lambda=0.3$)	43.44	29.09	27.00	25.83	25.23
M-Modes ($\delta=1$)	43.44	28.34	26.58	25.62	24.92
($\delta=3$)	43.44	28.34	26.28	24.98	24.12
($\delta=4$)	43.44	30.42	28.74	–	–
Min-Loss M-Best (1000)	39.09	28.34	26.91	25.58	25.06

$M=$	1	3	5	7	9
Alarm (37)					
M-Best	20.07	17.69	17.18	16.77	16.43
Diverse M-Best ($\lambda=0.1$)	20.07	16.36	15.18	14.71	13.99
($\lambda=0.3$)	20.07	17.25	17.02	16.72	16.44
M-Modes ($\delta=1$)	20.07	15.94	14.37	12.74	12.02
($\delta=3$)	20.07	15.86	13.87	12.92	12.11
Min-Loss M-Best (1000)	20.07	15.73	13.77	12.63	11.82
Hepar2 (70)					
M-Best	22.27	21.32	20.88	20.66	20.51
Diverse M-Best ($\lambda=0.1$)	22.27	20.99	20.02	19.84	19.72
($\lambda=0.3$)	22.27	21.27	21.24	21.24	21.24
M-Modes ($\delta=1$)	22.27	21.16	20.88	20.62	20.53
Min-Loss M-Best (1000)	22.26	20.95	20.24	19.89	19.62
Win95pts (76)					
M-Best	9.22	7.75	7.34	7.11	6.79
Diverse M-Best ($\lambda=0.1$)	9.22	8.08	7.57	7.18	6.81
($\lambda=0.3$)	9.22	8.15	8.12	8.08	8.08
M-Modes ($\delta=1$)	9.22	7.75	7.02	6.47	6.28
($\delta=3$)	9.22	7.84	6.99	6.62	6.49
Min-Loss M-Best (1000)	9.22	7.35	6.89	6.34	6.28

M-Modes' generally works well at $\delta = 3$. We omit other larger δ size, while we added $\delta = 4$ for Child as reference, they are not as good as $\delta = 3$. The fact that δ value is integer somehow makes it lack of capability to reach best performance.

The following is detailed discussion on each network respectively:

Asia is a small network has only 8 variables, and each variable has 2 labels. So, there are only $2^8 = 256$ different labelings. We observe the M-Best solutions probabilities, and found top 5 solutions have probabilities: 0.29, 0.20, 0.15, 0.11, and 0.05. The probability distribution is very steep, so that it much prefer top higher probability solutions than diversity. That also explained why the M-Best has fairly good performance.

Sachs is another small network with only 11 variables and each variable have 3 labels. The results are general worse than Asia's, because the probability distribution is obviously small and flat. The top five solutions' probabilities are: 0.019, 0.016, 0.013, 0.012, and 0.011.

Child is a medium size network, and has 20 variables. The label size is from 2 to 6. The results show worse performances on all methods. It is not surprising since the top 10-Best solutions are from 5.8×10^{-3} to 2.1×10^{-3}, which is both low and flat with respect to a not large model. We can see that as M-Modes is increasing diversity, its error rates are getting lower at $\delta = 3$, and then going up from $\delta = 4$. This is a typical phenomenon at diverse inference methods that they need to find this "dent" by cross-validation.

Alarm is another medium size network with 37 variables, and 2 to 4 label size. It has a fairly well general accuracy on all tests.

Hepar2 is a large network with 70 variables. It has label size 2 to 4. Because top M-Best solutions are low and flat, top 10 solutions' probabilities are from 7.8×10^{-8} to 5.8×10^{-8} for example, top 1000 or 10000 still cannot fully converge to the exact solution. Meanwhile, few of

modes (we can only compute 4 modes for $\delta = 2$) also indicates this distribution is smooth such that M-Modes method cannot utilize its design to have many diverse solutions.

Win95pts is another large network with 76 variables. It has label size 2 for all of the variables. We find that the top 3-Best's probabilities are 0.05, 0.033, and 0.015. For such a large network, they are high and steep. This explains why most of the results work well on *Win95pts*. Its probability distribution is high and steep. Our Min-Loss M-Best method works much better than others over all chosen m values.

5.2 Top M and Running Time

In the last two experiments, we use a medium network *Alarm* and choose $M = 5$ to quantitatively observe (1) how the m chosen and (2) how the running time impact the performances of Min-Loss M-Best.

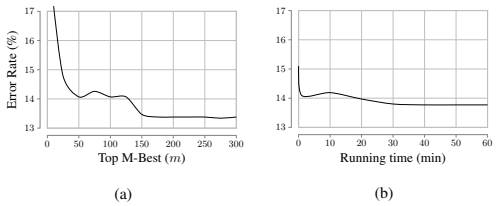


Figure 6: Experimental results for Min-Loss M-Best of descending error rates (%) on different settings for network Alarm $M = 5$

First experiment, we exact solve the problem from different top m-best solutions from 10 to 300. See Figure 6 (Left). The error rates quickly dive from 17.18 reaching a local minimum at about top 50 and go a bit higher at top 75, then keep going down. When the number of top solutions is large enough to depict the distribution, including more top solutions may not guarantee a lower error rates, it could oscillate a bit. But globally the trend does slowly converge.

The running time trend is similar. See Figure 6 (Right). It quickly gets a fairly low error rate solution, but keep staying and sometimes oscillate at that position. A longer running time does help to find a even better solution, but most of the time, it stays and does not descend.

6. Concluding Remarks

We developed and evaluated a fundamental formulation for multiple inference, as a Bayesian method approach, by directly optimizing oracle accuracy via expected loss. The biggest advantage of the new method is that it is parameter free, in contrast to other MAP estimation diverse approaches. We demonstrated that this idea is clearly effective.

Although promising, the proposed methods are currently restricted by approximately using top M-Best solutions to simulate the whole distribution. If the model is very large, top M-Best solutions may not be able to depict the whole distribution, as "long tail" phenomena happens. But expanding beyond top M solutions will lead this problem to be much harder to solve. More advanced techniques are yet to be developed. For example when using nodewise distance measures, it is worth investigating whether we can continue utilizing this splittable property (as Theorem 1) either to prune some values or divide and conquer the problem in a much more intelligent way.

References

D. Batra, P. Yadollahpour, A. Guzman-Rivera, and G. Shakhnarovich. Diverse M-best solutions in markov random fields. *Computer Vision–ECCV 2012*, pages 1–16, 2012.

I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. *The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks*. Springer, 1989.

C. Chen, V. Kolmogorov, Y. Zhu, D. Metaxas, and C. H. Lampert. Computing the M most probable modes of a graphical model. In *International Conf. on Artificial Intelligence and Statistics (AISTATS)*, 2013.

C. Chen, C. Yuan, Z. Ye, and C. Chen. Solving m-modes in loopy graphs using tree decompositions. In *International Conference on Probabilistic Graphical Models*, pages 145–156, 2018.

R. Dechter, N. Flerova, and R. Marinescu. Search algorithms for m best solutions for graphical models. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 1895–1901. AAAI Press, 2012. URL `http://dl.acm.org/citation.cfm?id=2900929.2900996`.

D. Dey, V. Ramakrishna, M. Hebert, and J. Andrew Bagnell. Predicting multiple structured visual interpretations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2947–2955, 2015.

N. Flerova, R. Marinescu, and R. Dechter. Searching for the m best solutions in graphical models. *Journal of Artificial Intelligence Research*, 55:889–952, 2016.

M. Fromer and A. Globerson. An LP view of the M-best MAP problem. *Advances in Neural Information Processing Systems*, 22:567–575, 2009.

M. Fromer and C. Yanover. Accurate prediction for atomic-level protein design and its application in diversifying the near-optimal sequence space. *Proteins: Structure, Function, and Bioinformatics*, 75(3):682–705, 2009.

K. Gimpel, D. Batra, C. Dyer, and G. Shakhnarovich. A systematic exploration of diversity in machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1100–1111, 2013.

A. Kirillov, B. Savchynskyy, D. Schlesinger, D. Vetrov, and C. Rother. Inferring M-Best diverse labelings in a single one. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015.

S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–224, 1988.

E. L. Lawler. A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management Science*, 18(7):401–405, 1972. ISSN 00251909, 15265501. URL `http://www.jstor.org/stable/2629357`.

F. Li, J. Carreira, and C. Sminchisescu. Object recognition as ranking holistic figure-ground hypotheses. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1712–1719. IEEE, 2010.

K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

D. Nilsson. An efficient algorithm for finding the m most probable configurations in probabilistic expert systems. *Statistics and Computing*, 8(2):159–173, 1998.

A. Onisko. Probabilistic causal models in medicine: Application to diagnosis of liver disorders. In *Ph. D. dissertation, Inst. Biocybern. Biomed. Eng., Polish Academy Sci., Warsaw, Poland*, 2003.

J. Pearl. Probabilistic reasoning in intelligent systems. palo alto. *Morgan Kaufmann. PEAT, J., VAN DEN BERG, R., & GREEN, W.(1994). Changing prevalence of asthma in australian children. British Medical Journal*, 308:1591–1596, 1988.

K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

D. J. Spiegelhalter, A. P. Dawid, S. L. Lauritzen, R. G. Cowell, et al. Bayesian analysis in expert systems. *Statistical science*, 8(3):219–247, 1993.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

P. Yadollahpour, D. Batra, and G. Shakhnarovich. Discriminative re-ranking of diverse segmentations. *Proc. of IEEE Conference on CVPR*, 2013.