Appendix for MAP Inference in Chains Using Column Generation

David Belanger, Alexandre Passos, Sebastian Riedel, Andrew McCallum

A Extensions of the Algorithm

The column-generation algorithm is fairly general, and can be used for other purposes than those described so far.

A.1 General Column Generation Algorithm for Trees

We can extend this algorithm to tree graphical models by using an analogous reduced-cost expression where information is incorporated from all leaves of the tree (instead of both sides of the chain). This gives us the following expression for the reduced cost

$$\begin{aligned} N_l R_{ij}(x_i, x_j) &= N_l \tau_{ij}(x_i, x_j) + N_{ij}\theta_i(x_i) + N_{ji}\theta_j(x_j) \\ &\quad + N_{ij}\left(\sum_{k\in N(i)-j} \lambda_{ki}(x_i) - \lambda_{ij}(x_j)\right) + N_{ji}\left(\sum_{k\in N(j)-i} \lambda_{kj}(x_j) - \lambda_{ji}(x_i)\right) \end{aligned}$$

where N_l is the number of leaves in the tree and N_{ij} is the number of leaves reachable from node i without passing through j. The validity of this expression is proved in section C.

A.2 Approximate Inference using Duality Gap

Maximizing a problem's Lagrangian with respect to fixed dual variables always provides an upper bound on the obtainable primal objective, as this maximization may yield a primal infeasible setting [1]. In section D, we derive the following upper bound on the optimal solution of the unrestricted problem:

$$\sum_i \max_{x_i, x_{i+1}} R_i(x_i, x_{i+1})\mu_i(x_i, x_{i+1}) + \frac{1}{2}\max_{x_n}\left[\alpha_n(x_n) + \theta_n(x_n)\right] + \frac{1}{2}\max_{x_0}\left[\beta_0(x_0) + \theta_0(x_0)\right]$$

By comparing this upper bound with the current restricted problem optimum, we can terminate if the gap is small. This provides an approach to *approximate* inference in chains that is qualitatively different than beam search because it specifies the accuracy, rather than speed, in advance. One can compute this bound using beam search messages, but brute force computation is as expensive as exact inference. Therefore, an independent use of our reduced cost oracle is to enable tractable computation of a duality gap while doing beam search. However, even with this gap, it is unclear how to react if the it is too large. Increasing the beam-width is substantially more expensive than growing the domain of one edge in our algorithm. Without a backward beam, it would also fail to consider global information from both ends of the chain, and perhaps overestimate the gap.

A.3 Implementing a more efficient 0/1 loss oracle and doing k-best MAP

When doing structured SVM learning it is necessary to perform loss-augmented inference during training [2]. For the Hamming loss, which factorizes into local scores, a loss-augmented oracle can employ any MAP inference routine. The 0/1 loss, however, does not factorize across the chain, as it is 1 for all settings of the variables which disagree at all with the ground truth. A simple way of implementing loss-augmented inference for the 0/1 loss is to perform two-best MAP inference, and report a margin violation whenever either the best-scoring solution disagrees with the ground truth or the second-best solution has a score which is closer to the ground truth than a specified margin.

Even though the known LP formulations of the k-best MAP inference problem have constraints which do not lead to efficient message-passing algorithms [3], we can construct an efficient 0/1-loss oracle using column generation by adding all edges with reduced cost at least as big as $-m$, where m is the margin.

Theorem 1. *If we add to the restricted LP all edge variable settings with reduced cost larger than $-\gamma$, k-best Viterbi inference using only the edges in the restricted problem is guaranteed to return all of the top-k assignments whose score is no more than γ worse than the MAP score.*

The proof is in section E.

A.4 Training Models that will Decode Quickly

Our algorithm is a member of the family of *energy aware* inference methods [4], in which the algorithm's behavior adapts at runtime to the model's coefficients. In practice, the efficiency of our algorithm is dominated by the reduced cost oracle, described in Figure 1, which depends on how many edge settings can be pruned before evaluating exact reduced costs. Pruning depends on the tightness of the the upper bounds U_i and U'_i for best possible reduced cost achievable for a given node setting, and these depend closely on bounds on the rows and columns of the transition matrix.

At training time, we can regularize a model to encourage the tightness of these bounds. For example, we can penalize the magnitude of transition scores more than local scores. This provides an additional method for making exact inference fast, by learning parameters for which it is efficient to perform exact inference, instead of constraining inference to be approximate at test time.

B Proof of correctness of our reduced-cost oracle

For the oracle to be correct all that is necessary is that if there is an assignment to an edge variable with positive reduced cost it should be eventually returned. To prove that assume by contradiction that such an assignment exists and the reduced cost oracle did not return it. Note that for this to have happened then this assignment must have had either its left or right state pruned, because the algorithm explicitly evaluates the reduced cost for all non-pruned assignments.

Let's, then, without loss of generality, assume that the value of the left variable was pruned. Then, we know that $S_i^+(x_i) + \left(\max_{x_j} S_i^-(x_j)\right) + 2U_\tau(x_i,\cdot) \leq 0$, as otherwise the pruning would not have happened. However, $S_i^+(x_i) + \left(\max_{x_j} S_i^-(x_j)\right) + 2U_\tau(x_i,\cdot)$ is at least as large as the reduced cost of any edge assignment that assigns x_i to variable i, so if that is negative then so is the reduced cost, which leads us to a contradiction and finishes the proof.

C Derivation of Reduced Cost Expression for Trees

The LP version of the MAP problem in trees is

$$\begin{aligned}
\textbf{max}. \quad & \sum_{i,x_i} \mu_i(x_i)\theta_i(x_i) + \sum_{ij\in E}\sum_{x_i,x_j} \mu_{ij}(x_i,x_j)\tau_{ij}(x_i,x_j) \\
\textbf{s.t}. \quad & \sum_{x_i} \mu_i(x_i) = 1 \\
& \sum_{x_i} \mu_{ij}(x_i,x_j) = \mu_j(x_j) \\
& \sum_{x_j} \mu_{ij}(x_i,x_j) = \mu_i(x_i)
\end{aligned} \tag{C.1}$$

Following Wainwright and Jordan [5], we choose an arbitrary node as the root of the tree and rewrite C.1 as

$$\begin{aligned}\textbf{max.}\quad & \sum_{x_1} \mu_1(x_1)\theta_1(x_1) + \sum_{ij\in E}\sum_{x_i,x_j} \mu_{ij}(x_i,x_j)(\tau_{ij}(x_i,x_j)+\theta_j(x_j)) \\ \textbf{s.t.}\quad & \sum_{x_1}\mu_1(x_1) = 1 \\ & \sum_{x_i}\mu_{ij}(x_i,x_j) = \sum_{x_k}\mu_{jk}(x_j,x_k) \qquad \forall k\in N(j)-i, \forall i,j,x_j.\end{aligned} \tag{C.2}$$

This is done by assigning the local score of each node, besides the root, to its parent transition edge.

From this LP you can derive the following Lagrangian,

$$\begin{aligned}L(\mu,\lambda,T) = & \sum_{x_1}\mu_1(x_1)\theta_1(x_1) + \sum_{ij\in E}\sum_{x_i,x_j}\mu_{ij}(x_i,x_j)(\tau_{ij}(x_i,x_j)+\theta_j(x_j)) + T\left(\sum_{x_1}\mu_1(x_1)-1\right) \\ & + \sum_{ij\in E}\sum_{x_j}\sum_{k\in N(j)-i}\lambda_{kj}(x_j)\left(\sum_{x_k}\mu_{jk}(x_j,x_k) - \sum_{x_i}\mu_{ij}(x_i,x_j)\right)\end{aligned} \tag{C.3}$$

whose terms can be rearranged to give rise to the following expression with reduced costs

$$\begin{aligned}L(\mu,\lambda,T) = & \sum_{x_1}\mu_1(x_1)\left(\theta_1(x_1) - \sum_{j\in N(1)}\lambda_{ij}(x_1) - T\right) \\ & + \sum_{ij\in E}\sum_{x_i,x_j}\mu_{ij}(x_i,x_j)\left(\tau_{ij}(x_i,x_j)+\theta_j(x_j) + \sum_{k\in N(j)-i}\lambda_{kj}(x_j) - \lambda_{ji}(x_i)\right)\end{aligned} \tag{C.4}$$

By setting the maximum reduced cost of the pairwise marginals at each edge to zero we get the standard max-product message-passing updates on trees,

$$\lambda_{ji}(x_i) = \max_{x_j}\tau_{ij}(x_i,x_j) + \theta_j(x_j) + \sum_{k\in N(j)-i}\lambda_{kj}(x_j) \tag{C.5}$$

Note that the value of these messages depends only on the fact that the root of the tree is closer to i than it is to j; otherwise the message going across this edge would be analogous, except going in the other direction.

This shows that if we picked each leaf in turn as the root, computed the Lagrangian, and averaged them, we'd get, at each edge, only two distinct values for the dual variables associated with it, depending on whether the root is closer to i or to j from that edge. The average lagrangian, then, would have a value equal to

$$\begin{aligned}L(\mu,\lambda,T) = & \frac{1}{N_l}\sum_{\text{leaf } l}\sum_{x_k}\mu_l(x_l)\left(\theta_l(x_l) - \sum_{j\in N(1)}\lambda_{ij}(x_1) - T\right) \\ & + \sum_{ij\in E}\frac{1}{N_i+N_j}\sum_{x_i,x_j}\mu_{ij}(x_i,x_j)\Bigg((N_i+N_j)\tau_{ij}(x_i,x_j) + N_i\theta_i(x_i)N_j\theta_j(x_j) \\ & + N_i\left(\sum_{k\in N(j)-i}\lambda_{kj}(x_j) - \lambda_{ji}(x_i)\right) + N_j\left(\sum_{k\in N(i)-j}\lambda_{ki}(x_i) - \lambda_{ij}(x_j)\right)\Bigg)\end{aligned} \tag{C.6}$$

where N_l is the number of leaves in the tree, N_i the number of leaves reachable from node i.

From this Lagrangian we can then derive a primal LP problem as follows

$$\begin{aligned}
\textbf{max}.\quad & \frac{1}{N_l}\sum_l\sum_{x_l}\mu_l(x_l)\theta_l(x_l)+\sum_{ij}\frac{1}{N_i+N_j}(N_l\tau_{ij}(x_i,x_j)+N_i\theta_i(x_i)+N_j\theta_j(x_j)) \\
\textbf{s.t.}\quad & \sum_{x_l}\mu_l(x_l)=1 && \forall\text{leaf } l \\
& \sum_{k\in N(i)-j} N_k\sum_{x_k}\mu_{ki}(x_k,x_i)=N_i\sum_{x_j}\mu_{ij}(x_i,x_j) && \forall i,x_i
\end{aligned}\tag{C.7}$$

Note that the constraint here is equivalent to the sum of many marginalization constraints, as

$$\sum_{k\in N(i)-j} N_k=N_i,$$

as the number of leaves reachable through all descendants of i has to be equal to the number of leaves reachable through i.

Also note that this LP is a generalization for trees of the LP for chains presented in the main text.

The reduced cost in such an LP, for each edge ij, is

$$\begin{aligned}
N_lR_{ij}(x_i,x_j)=(N_i+N_j)\tau_{ij}(x_i,x_j)+N_i\theta_i(x_i)+N_j(\theta_j(x_j)) \\
+N_i\left(\sum_{k\in N(i)-j}\lambda_{ki}(x_i)-\lambda_{ij}(x_j)\right) \\
+N_j\left(\sum_{k\in N(j)-i}\lambda_{kj}(x_j)-\lambda_{ji}(x_i)\right)
\end{aligned}\tag{C.8}$$

And, analogously to the chain case, it is easy to see that setting all dual variables to the fixed points of the max-product messages is a sufficient but not necessary condition for dual feasibility.

D Proof of the Upper Bound formula for Chains

It is useful to work with a slightly different formulation of LP (10) from the paper. We add the explicit, but redundant, constraint that only one setting can be used for each edge (the final constraint below).

$$\begin{aligned}
\textbf{max}.\quad & \sum_{i,x_i,x_{i+1}}\mu_i(x_i,x_{i+1})\left(\tau(x_i,x_{i+1})+\frac{1}{2}\theta_i(x_i)+\frac{1}{2}\theta_{i+1}(x_{i+1})\right) \\
\textbf{s.t.}\quad & \sum_{x_n}\mu_n(x_n,\cdot)=1 && (N^+) \\
& \sum_{x_1}\mu_0(\cdot,x_1)=1 && (N^-) \\
& \sum_{x_{i-1}}\mu_{i-1}(x_{i-1},x_i)=\sum_{x_{i+1}}\mu_i(x_i,x_{i+1}) && (\alpha_i(x_i)) \\
& \sum_{x_{i+1}}\mu_i(x_i,x_{i+1})=\sum_{x_{i-1}}\mu_{i-1}(x_{i-1},x_i) && (\beta_i(x_i)) \\
& \sum_{x_i,x_{i+1}}\mu_i(x_i,x_{i+1})=1 && (N_i)
\end{aligned}\tag{D.1}$$

We form a Lagrangian by relaxing all but the edge normalization constraints that we just introduced and performing the same multiplication of some constraints by 2 or $\frac{1}{2}$ done in the paper.

$$L(\mu, \alpha, \beta, N^+, N^-) = \sum_i \sum_{x_i, x_{i+1}} R_i(x_i, x_{i+1}) \mu_i(x_i, x_{i+1}) + \frac{1}{2}(N^+ + N^-), \tag{D.2}$$

where the reduced cost of the $\mu_i(x_i, x_{i+1})$ variables is the same used in the paper, i.e. equation (12).

To evaluate the Lagrangian with respect to fixed messages α and β, we need to choose values for N^+ and N^- and then maximize over the primal variables μ.

Maximizing the Lagrangian over the primal variables subject to the edge normalization constraints is simple. It's optimum is the following dual objective:

$$D(\alpha, \beta, N^+, N^-) = \sum_i \max_{x_i, x_{i+1}} R_i(x_i, x_{i+1}) \mu_i(x_i, x_{i+1}) + \frac{1}{2}(N^+ + N^-), \tag{D.3}$$

To obtain the least upper bound, we minimize $D(\alpha, \beta, N^+, N^-)$ over N^+ and N^-. We know from the LP dual presented in the paper that the dual constraints involving N^+ are that

$$N^+ - \alpha_n(x_n) \geq \theta_n(x_n) \ \forall x_n \tag{D.4}$$

The constraints involving N^- are similar, except at the beginnng of the chain. We obtain the following upper bound:

$$U(\alpha, \beta) = \sum_i \max_{x_i, x_{i+1}} R_i(x_i, x_{i+1}) \mu_i(x_i, x_{i+1}) + \frac{1}{2} \max_{x_n} [\alpha_n(x_n) + \theta_n(x_n)] + \frac{1}{2} \max_{x_0} [\beta_0(x_0) + \theta_0(x_0)] \tag{D.5}$$

E Proof of Correctness for 0/1 Loss Oracle

We seek to prove the following statement:

If we add to the restricted LP all edge variable settings with reduced cost larger than $-\gamma$, k-best Viterbi inference using only the edges in the restricted problem is guaranteed to return all of the top-k assignments with score equal to $MAP - \gamma$ or larger, where MAP is the score of the MAP assignment.

While the notion of 2-best setting is not well-defined in terms of the LP (because it can take on non-integral values), the one-to-one correspondence between settings of the graphical model variables and feasible integral settings of the LP variables allows us to reason about the properties of specific parts of a two-best solution.

Let μ^*, λ^* be a primal-dual optimal pair for a MAP inference LP, and let $D_1, \ldots, D_n$ be the restricted domains of the primal variables. Let $S(\mu)$ refer to the LP objective for μ and $L(\mu, \lambda)$ refer to the Lagrangian score for μ, λ.

We'll use the subscript v to index the LP variables. Let $R_v(\mu_v, \lambda)$ refer to the reduced cost of LP variable v w.r.t dual variable vector λ.

Suppose, for the sake of contradiction, that there exists a setting of the graphical model variables such that its corresponding integral LP variable setting $\tilde{\mu}$ is such that $S(\mu^*) - S(\tilde{\mu}) < \gamma$ and some component of $\tilde{\mu}$ isn't in the instantiated pairwise domains. I.e. $\exists v \ s.t. \ \tilde{\mu}_v \notin D_v$. Note that by defining $\tilde{\mu}$ in terms of a setting to the graphical model variables, we are specifying that $\tilde{\mu}$ satisfies the primal constraints by construction.

Since $\tilde{\mu}$ satisfies the primal constraints, $S(\tilde{\mu}) = L(\tilde{\mu}, \lambda^*)$. We also have that $S(\mu^*) = L(\tilde{\mu}, \lambda^*)$ because μ^*, λ^* is a primal-dual pair. Therefore, we have the condition that $L(\tilde{\mu}, \lambda^*) - L(\tilde{\mu}, \lambda^*) < \gamma$. Now, let's rewrite the terms of the Lagrangians in terms of reduced costs. Let V be the index set for the components of μ_v:

$$\sum_{v \in V} R_v(\mu_v^*, \lambda^*) \mu_v^* - \sum_{v \in V} R_v(\tilde{\mu}_v, \lambda^*) \tilde{\mu}_v < \gamma \tag{E.1}$$

Since $\tilde{\mu}$ is integral, we know that $\tilde{\mu}_v$ is either 0 or 1. The same is true for μ_v^* because we know the LP is tight. Therefore, we can cancel out the sums over the subset of V where $\tilde{\mu}_v = \mu_v^*$. We next split up this index set of places where they disagree into two sets, S^+, where $\mu^*_v = 1$ and $\tilde{\mu}_v = 0$, and S^-, where $\mu^*_v = 0$ and $\tilde{\mu}_v = 1$:

$$\sum_{v \in S^+} R_v(\mu^*_v, \lambda^*)\mu^*_v - \sum_{v \in S^-} R_v(\tilde{\mu}_v, \lambda^*)\tilde{\mu}_v < \gamma \tag{E.2}$$

By the optimality of μ^*, we know that every term in the left hand sum is zero. Therefore, we have that:

$$\sum_{v \in S^-} R_v(\tilde{\mu}_v, \lambda^*)\tilde{\mu}_v + \gamma > 0 \tag{E.3}$$

Next, we split S^- up further into two sets. Let S^-_I refer to the index set of components $\tilde{\mu}_v$ with $\tilde{\mu}_v \in D_v$ and S^-_O refer to components $\tilde{\mu}_v$ with $\tilde{\mathbf{x}}_v \notin D_v$.

By the initial assumptions of the proof, we have that S^-_O is nonempty. We actually have that $|S^-_O| \geq 2$. If $|S^-_O| = 1$, then there's no way that the primal constraints could be satisfied by $\tilde{\mu}$ because LP variables for adjacent edges in the graphical model must agree on the setting of the node where they intersect.

For $v \in S^-_I$, we have that $R_v(\tilde{\mu}_v, \lambda^*) < 0$, by the fact that $\tilde{\mu}_v$ isn't used in the LP optimum. For $v \in S^-_O$, we have that $R_v(\tilde{\mu}_v, \lambda^*) < -\gamma$, by the assumptions of the proof. Therefore, we have:

$$\begin{aligned}
\sum_{v \in S^-} R_v(\tilde{\mu}_v, \lambda^*)\tilde{\mu}_v + \gamma &= \sum_{v \in S^-} R_v(\tilde{\mu}_v, \lambda^*) + \gamma && \text{(E.4)}\\
&= \sum_{v \in S^-_I} R_v(\tilde{\mu}_v, \lambda^*) + \sum_{v \in S^-_O} R_v(\tilde{\mu}_v, \lambda^*) + \gamma && \text{(E.5)}\\
&< \sum_{v \in S^-_O} R_v(\tilde{\mu}_v, \lambda^*) + \gamma && \text{(E.6)}\\
&< -\gamma |S^-_O| + \gamma && \text{(E.7)}\\
&< -2\gamma + \gamma && \text{(E.8)}\\
&< 0 && \text{(E.9)}
\end{aligned}$$

But we had in equation (E.3) that $\sum_{v \in S^-} R_v(\tilde{\mu}_v, \lambda^*)\tilde{\mu}_v + \gamma > 0$. Therefore, we have a contradiction, and the proof is complete.

Note that if one needs k-best assignments for some other purpose and the value of r is not known *a priori* it can be determined by first guessing any value of r (including 0), running k-best decoding using only the edges originally added to the restricted LP, computing the score of the k-th solution returned, and set r to be the difference between that score and the optimal score, as then one is guaranteed to find the true k best settings possibly using a small subset of the settings of the edge variables in the graphical model.

References

[1] S.P. Boyd and L. Vandenberghe. *Convex optimization.* Cambridge Univ Pr, 2004.

[2] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(2):1453, 2006.

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

[4] D. Tarlow, D. Batra, P. Kohli, and V. Kolmogorov. Dynamic tree block coordinate ascent. In *ICML*, pages 113–120, 2011.

[5] 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.