Partition Functions from Rao-Blackwellized Tempered Sampling: Supplemental Material

A. Mixed $\hat{Z}$ Updates

We can generalize our Rao-Blackwellized maximum likelihood interpretation in Section 2.3 to situations in which $\hat{Z}$ is not a fixed set of quantities for all samples. Under these conditions, we can no longer use the update in (12). However, we can easily find the Rao-Blackwellized log-likelihood, assuming independent β_k samples. Approximately independent samples can be obtained by sub-sampling with a rate determined by the autocorrelation of sampled β. We empirically found that varying $\hat{Z}$ at late stages did not have a large effect on estimates.

Assume we have samples $\{x^{(i)}, \beta^{(i)}\}$, with $\beta|x^{(i)}$ sampled using estimates $\hat{\mathbf{Z}}^{(i)} = (\hat{Z}_k^{(i)})_{k=1}^K$. Then our Rao-Blackwellized log-likelihood is the following

$$L\left[\mathbf{Z}; \{\hat{\mathbf{Z}}^{(i)}\}_{i=1}^N\right] = \sum_{i=1}^N \sum_{k=2}^K \log Z_k q(\beta_k | x^{(i)}; \hat{\mathbf{Z}}^{(i)}) - \sum_{i=1}^N \log \left(\sum_{k'=1}^K r_{k'} Z_{k'} / \hat{Z}_{k'}^{(i)} \right),$$

where

$$q(\beta_k | x; \hat{\mathbf{Z}}^{(i)}) = \frac{f_k(x) r_k / \hat{Z}_k^{(i)}}{\sum_{k'=1}^K f_{k'}(x) r_{k'} / \hat{Z}_{k'}^{(i)}}.$$

Note that this expression is concave in $\log Z$ and can be solved efficiently using the generalized gradient descent methods of (Carlson et al., 2015a; 2016). The total computational time of this approach will scale $\mathcal{O}(K)$, whereas the Newton-Raphson method proposed in MBAR would scale $\mathcal{O}(K^3)$ per-iteration. It is not clear how the number of iterations required in Newton-Raphson will scale, and could potentially have a worse dependence on K.

B. Bias and Variance derivations

A Taylor expansion of $\log \hat{Z}_k^{\text{RTS}}$, using (11)-(12) and $\log(1+x) \simeq x - x^2/2$, gives

$$\log \hat{Z}_k^{\text{RTS}} \approx \log Z_k + \frac{\Delta c_k}{q_k} - \frac{\Delta c_1}{q_1} - \frac{(\Delta c_k)^2}{2q_k^2} + \frac{(\Delta c_1)^2}{2q_1^2}$$

where $q_k = q(\beta_k)$ and $\Delta c_k = \hat{c}_k - q_k$. Taking expectations, and replacing q_k by its estimate $\hat{c}_k$, gives

$$\mathbb{E}\left[\log \hat{Z}_k^{\text{RTS}}\right] - \log Z_k \approx \frac{1}{2}\left[\frac{\sigma_1^2}{\hat{c}_1^2} - \frac{\sigma_k^2}{\hat{c}_k^2}\right], \tag{25}$$

and

$$\text{Var}[\log \hat{Z}_k^{\text{RTS}}] \approx \frac{\sigma_1^2}{\hat{c}_1^2} + \frac{\sigma_k^2}{\hat{c}_k^2} - \frac{2\sigma_{1k}}{\hat{c}_k \hat{c}_1} \tag{26}$$

where $\sigma_1^2 = \text{Var}[\hat{c}_1]$, $\sigma_k^2 = \text{Var}[\hat{c}_k]$, and $\sigma_{1k} = \text{Cov}[\hat{c}_1, \hat{c}_k]$.

From the CLT, the asymptotic variance of $\hat{c}_k$ is

$$Var(\hat{c}_k) = \frac{Var_q(q(\beta_k|x))a_k}{N}, \tag{27}$$

where the factor

$$a_k = 1 + 2\sum_{i=1}^{\infty} \text{corr}\left[q(\beta_k|x^{(0)}), q(\beta_k|x^{(i)})\right] \tag{28}$$

takes into account the autocorrelation of the Markov chain. But estimates of this sum from the MCMC samples are generally too noisy to be useful. Alternatively, $Var[\hat{c}_k]$ could simply be estimated from $\hat{c}_k$ estimates on many parallel MCMC chains.

C. RTS and TI-RB Continuous β Equivalence

We want to show the relationship mentioned in (22), which we repeat here:

$$\log\left(\frac{\hat{Z}_K}{Z_1}\right)^{(RTS)} = \int_0^1 \frac{d}{d\beta}\left(\log \hat{c}_\beta - \log r_\beta + \log \hat{Z}_\beta\right) d\beta, \\ = \int_0^1 \frac{\sum_i q(\beta|x_i)\Delta_{x_i}}{\sum_j q(\beta|x_j)} d\beta.$$

Note that we can write the statistics c_k as

$$c_k = \sum_{i=1}^N q(\beta_k|x_i) \\ = \sum_{i=1}^N \frac{\exp\left(\beta_k \Delta_{x_i} + \log r_k - \log \hat{Z}_k\right)}{\sum_{k'=0}^K \exp\left(\beta_{k'}\Delta_{x_i} + \log r_{k'} - \log \hat{Z}_{k'}\right)}$$

The continuous version of this replaces the index k by β, and

$$\begin{aligned} c_\beta &= \sum_{i=1}^{N} q(\beta|x_i) \\ &= \sum_{i=1}^{N} \frac{\exp\left(\beta\Delta_{x_i} + \log r_\beta - \log \hat{Z}_\beta\right)}{\int_0^1 \exp\left(\alpha\Delta_{x_i} + \log r_\alpha - \log \hat{Z}_\alpha\right) d\alpha} \end{aligned}$$

The continuous form of the RTS estimator can be written as an integral:

$$\begin{aligned} \log \frac{Z_K}{Z_1} &= \left(\log c_\beta - \log r_\beta + \log \hat{Z}_\beta\right)\Big|_{\beta=1} \\ &\quad - \left(\log c_\beta - \log r_\beta + \log \hat{Z}_\beta\right)\Big|_{\beta=0} \\ &= \int_0^1 \frac{d}{d\beta}\left(\log c_\beta - \log r_\beta + \log \hat{Z}_\beta\right) d\beta \end{aligned} \tag{29}$$

We first analyze the derivative of c_β, which is

$$\begin{aligned} &\frac{d}{d\beta}\log c_\beta \\ =&\frac{d}{d\beta}\log \sum_{i=1}^{N} \frac{\exp\left(\beta\Delta_{x_i} + \log r_k - \log \hat{Z}_k\right)}{\int_0^1 \exp\left(\alpha\Delta_{x_i} + \log r_\alpha - \log z_\alpha\right) d\alpha} \\ =&\frac{1}{\sum_{i=1}^{N} \frac{\exp\left(\beta\Delta_{x_i} + \log r_\beta - \log \hat{Z}_\beta\right)}{\int_0^1 \exp\left(\alpha\Delta_{x_i} + \log r_\alpha - \log \hat{Z}_\alpha\right) d\alpha}} \\ &\times \sum_{i=1}^{N} \frac{\exp\left(\beta\Delta_{x_i} + \log \frac{r_\beta}{\hat{Z}_\beta}\right) \frac{d}{d\beta}\left(\beta\Delta_{x_i} + \log \frac{r_\beta}{\hat{Z}_\beta}\right)}{\int_0^1 \exp\left(\alpha\Delta_{x_i} + \log r_\alpha - \log \hat{Z}_\alpha\right) d\alpha} \\ =&\sum_i \frac{q(\beta|x_i)\frac{d}{d\beta}\left(\beta\Delta_{x_i} + \log r_\beta - \log \hat{Z}_\beta\right)}{\sum_j q(\beta|x_j)} \\ =&\left[\sum_i \frac{q(\beta|x_i)}{\sum_j q(\beta|x_j)}\Delta_{x_i}\right] + \frac{d}{d\beta}(\log r_\beta - \log \hat{Z}_\beta) \end{aligned} \tag{30}$$

The last line follows since $\sum_{i=1}^{N} \frac{q(\beta|x_i)}{\sum_j q(\beta|x_j)} = 1$. The $\frac{d}{d\beta}(\log r_\beta - \log \hat{Z}_\beta)$ term in (29) and (30) simply cancel.

D. Similarity of RTS and MBAR

In this section, we elaborate on the similarity of the likelihood of MBAR and RTS. To prove this, we first restate the likelihood of MBAR given in (18):

$$\begin{aligned} L[\mathbf{Z}] = &\frac{1}{N}\sum_{i=1}^{N} \log\left(\sum_{k=1}^{K} \frac{n_k}{N} \exp(-\log Z_k + \beta_k\Delta_{x_i})\right) \\ &+ \sum_{k=1}^{N} \frac{n_k}{N} \log Z_k \end{aligned}$$

The partial derivative of this likelihood with respect to $\log Z_k$ is given by:

$$\begin{aligned} \frac{\partial L[\mathbf{Z}]}{\partial \log Z_k} = &\frac{n_k}{N} \\ &- \frac{1}{N}\sum_{i=1}^{N} \frac{\frac{n_k}{N}\exp(-\log Z_k + \beta_k\Delta_{x_i})}{\sum_{j=1}^{K} \frac{n_j}{N}\exp(-\log Z_j + \beta_j\Delta_{x_i})} \end{aligned} \tag{31}$$

Replacing $\frac{n_k}{N}$ with its expectation for all k gives

$$\begin{aligned} \frac{\partial L[\mathbf{Z}]}{\partial \log Z_k} = &q(\beta_k) \\ &- \frac{1}{N}\sum_{i=1}^{N} \frac{q(\beta_k)\exp(-\log Z_k + \beta_k\Delta_{x_i})}{\sum_{j=1}^{K} q(\beta_j)\exp(-\log Z_j + \beta_j\Delta_{x_i})} \end{aligned} \tag{32}$$

Noting that $q(\beta_k) \propto Z_k/\hat{Z}_k r_k$, we have

$$\begin{aligned} \frac{\partial L[\mathbf{Z}]}{\partial \log Z_k} &= q(\beta_k) \\ &\quad - \frac{1}{N}\sum_{i=1}^{N} \frac{\frac{Z_k}{\hat{Z}_k} r_k \exp(-\log Z_k + \beta_k\Delta_{x_i})}{\sum_{j=1}^{K} \frac{Z_j}{\hat{Z}_j} r_j \exp(-\log Z_j + \beta_j\Delta_{x_i})}, \\ &= q(\beta_k) \\ &\quad - \frac{1}{N}\sum_{i=1}^{N} \frac{\exp(-\log \hat{Z}_k + \beta_k\Delta_{x_i})}{\sum_{j=1}^{K} \exp(-\log \hat{Z}_j + \beta_j\Delta_{x_i})}, \\ &= q(\beta_k) - \frac{1}{N}\sum_{i=1}^{N} q(\beta_k|x_i), \\ &= q(\beta_k) - \hat{c}_k\,. \end{aligned} \tag{33}$$

Setting the partial derivative to 0 and substituting the definition of $q(\beta)$ into (33) gives a solution of

$$\frac{Z_k/\hat{Z}_k r_k}{\sum_{j=1}^{K} Z_j/\hat{Z}_j r_j} = \hat{c}_k, \tag{34}$$

which is identical to the RTS update in (12).

While RTS and MBAR give similar estimators, their intended use is a bit different. The MBAR estimator can be used whenever we have samples generated from a distribution at different temperatures, including both physical experiments where the temperature is an input and a tempered MCMC scheme. The RTS estimator requires a tempered MCMC approach, but in exchange has trivial optimization costs and improved empirical performance.

E. Adaptive HMC for tempering

Here we consider sampling from a continuous distribution using Hamiltonian Monte Carlo (HMC) (Neal, 2011). Briefly, HMC simulates Hamiltonian dynamics as a proposal distribution for Metropolis-Hastings (MH) sampling. In general, one cannot simulate exact Hamiltonian dynamics, so usually one uses the leapfrog algorithm, a first order discrete integration scheme which maintains the time-reversibility and volume preservation properties of Hamiltonian dynamics.

(Li et al., 2004) found using different step sizes improved sampling various multimodal distributions using random walk Metropolis proposal distributions. However, under their scheme, besides step sizes being monotonically decreasing in β, it is unclear how to set these step sizes. Additionally, in target distributions that are high-dimensional or have highly correlated variables, random walk Metropolis will work badly.

For most distributions of interest, as β decreases, $p(x|\beta)$ becomes flatter; thus, for HMC, we can expect the MH acceptance probability to decrease as a function of β, enabling us to take larger jumps in the target distribution when the temperature is high. As the stepsize ϵ of the leapfrog integrator gets smaller, the linear approximation of the solution to the continuous differential equations becomes more accurate, and the MH acceptance probability increases (for an infinitely small stepsize, the simulation is exact, and under Hamiltonian dynamics, the acceptance probability is 1). Thus, $p(\text{accept}|\epsilon)$ decreases with ϵ. Putting this idea together, we model $p(\text{accept}|\beta,\epsilon)$ as a logistic function for each $\beta \in \{0=\beta_1,...,\beta_J=1\}$

$$\text{logit}(p(\text{accept}|\beta,\epsilon)) = w_0^{(j)} + w_1^{(j)}\epsilon \tag{35}$$

Given data $\{(\beta^{(i)}, y^{(i)})\}_{i=1,\dots,N}$ with $y^{(i)}=1$ if the proposed sample i was accepted, and $y^{(i)}=0$ otherwise, we find

$$\begin{aligned} \max_{\{w^{(j)}\}} \quad & \sum_{j=1}^{J} h(w^{(j)}) \\ \text{s.t.} \quad & w_1^{(j)} \le 0 \\ & g(\beta_j,\epsilon) \le g(\beta_{j-1},\epsilon) \ \ \forall \epsilon \end{aligned} \tag{36}$$

where

$$\begin{aligned} h(w^{(j)}) = &\sum_{i:\beta^{(i)}=\beta_j} y^{(i)} \log(g(\beta^{(i)},\epsilon^{(i)})) \\ &+(1-y^{(i)})\log(1-g(\beta^{(i)},\epsilon^{(i)})) \end{aligned}$$

and

$$g(\beta_j,\epsilon) = p(\text{accept}|\beta_j,\epsilon) = \frac{1}{1+\exp(-(w_0^{(j)}+w_1^{(j)}\epsilon))}$$

The last constraint can be satisfied by enforcing $g(\beta_j,\epsilon_{\text{min}}) \le g(\beta_{j-1},\epsilon_{\text{min}})$ and $g(\beta_j,\epsilon_{\text{max}}) \le g(\beta_{j-1},\epsilon_{\text{max}})$, as doing so will ensure $g(\beta_j,\epsilon) \le g(\beta_{j-1},\epsilon)$ for all $\epsilon \in [\epsilon_{\text{min}},\epsilon_{\text{max}}]$. Before solving (36), we first run chains at fixed $\beta=0$ and $\beta=1$, running a basic stochastic optimization method to adapt each stepsize until the acceptance rate is close to the target acceptance rate, which we take to be 0.651, which is suggested by (Beskos et al., 2013). We take these stepsizes to be ϵ_{max} and ϵ_{min}, respectively. Once we have approximated $p(\text{accept}|\beta,\epsilon)$, choosing the appropriate proposal distribution given β is simple:

$$\hat{\epsilon}_{\text{opt}}(\beta_j) = \frac{\text{logit(p(acc))} - w_0^{(j)}}{w_1^{(j)}}$$

If $\hat{\epsilon}_{\text{opt}}$ is outside $[\epsilon_{\text{min}},\epsilon_{\text{max}}]$, we project it into the interval.

E.1. Example

Here we consider a target distribution of a mixture of two 10-dimensional Gaussians, each having a covariance of $0.5I$ separated in the first dimension by 5. Our prior distribution for the interpolating scheme is a zero mean Gaussian with covariance $30I$. The prior was chosen by looking at a one-dimensional projection of the target distribution and picking a zero-mean prior whose variance, σ^2, adequately covered both of the modes. The variance of the multidimensional prior was taken to be $\sigma^2 I$, and the mean to be $\mathbf{0}$. Our prior on temperatures was taken to be uniform. We compare the adaptive method above to simulation with a fixed step size, which is determined by averaging all of the step sizes, in an effort to pick the optimal fixed step size. The below figures show an improvement over the fixed step size in mixing and partition function estimation using our adaptive scheme.

We obtained similar improvements using random walk Metropolis by varying the covariance of an isotropic Gaussian proposal distribution. We note another scheme for discrete binary data may be used, where the number of variables in the target distribution to “flip”, as a function of temperature, is a parameter.

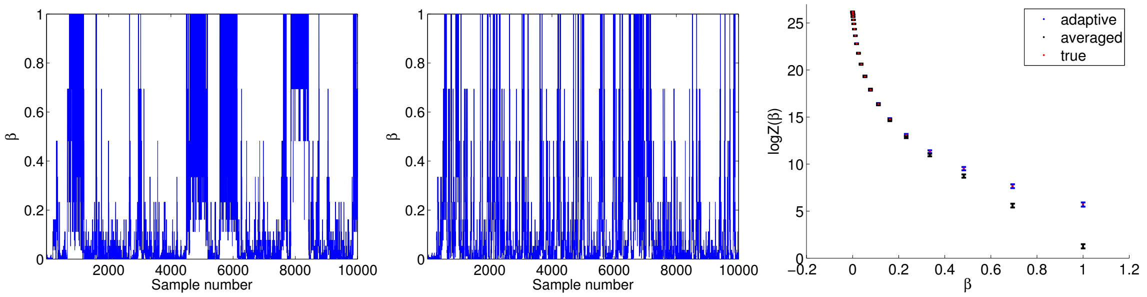

Figure 6. (Left) Mixing in β under the fixed step size. (Middle) Mixing in β under the adaptive scheme. (Right) Partition function estimates under the fixed step size and adaptive scheme after 10000 samples. Mixing in β using a fixed step size is visibly slower than mixing using the adaptive step size, which is reflected by the error in the partition function estimate.

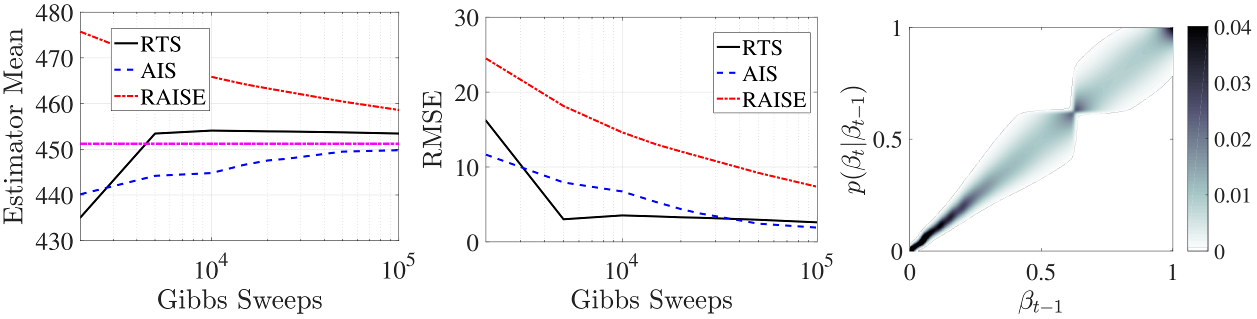

Figure 7. $\log Z$ estimates for an RBM with 784 visible units and 500 hidden units trained on the MNIST dataset when p_1 is a uniform distribution. (Left) The mean of the competing estimators. The magenta line gives truth. (Middle) The RMSE of the competing estimators. (Right) The empirical transition matrix on β clearly demonstrates that there is a "knot" in the temperature distribution that is prohibiting effective mixing and reducing estimator quality. This gives a simple diagnostic to analyze sampling results and mixing properties.

F. RBM $\log Z$ Estimates from a Uniform p_1

The choice of p_1 is known to dramatically affect the quality of log partition function estimates, and this was noted for RBMs in (Salakhutdinov & Murray, 2008). To demonstrate the comparative effect of a poor p_1 distribution on our estimator, we choose p_1 to have a uniform distribution over all binary patterns, and follow the same experimental setup as in Section 4.2. The quantitative results are shown in Figure 7 (Left) and (Middle). In this case all estimators behave significantly worse than when p_1 was intelligently chosen. We note that the initialization stage of RTS (see Section 2.4) takes significantly longer with this choice of p_1. Initially RTS decreases bias faster than AIS, but asymptotically they have similar behavior up to 10^5 Gibbs sweeps.

The poor performance of the estimators is due to a "knot" in the interpolating distribution caused by the mismatch between p_1 and p_K. This can be clearly seen in the empirical transition matrix over the inverse temperature β, shown in Figure 7 (Right). While we have limited our experiments to the interpolating distribution, a strength of our approach is that can naturally incorporate other possibilities that ameliorate these issues, such as moment averaging (Grosse et al., 2013) or tempering by subsampling (van de Meent et al., 2014), as mentioned in Section 2.1.

G. Estimating $q(\beta_k)$ from a transition matrix

Instead of estimating $q(\beta_k)$ by Rao-Blackwellizing via c_k in (9), it is possible to estimate $q(\beta_k)$ from the stationary distribution of a transition matrix. The key idea here is that the transition matrix accounts for the sampling structure used in MCMC algorithms, whereas c_k is derived using i.i.d. samples. Suppose that we have a transition sequence $\beta_1 \to \beta_2 \cdots \to \beta_N$. If $p(x|\beta)$ is an exact Gibbs sampler, then this is a Markov transition, since

$$\begin{aligned} p(\beta_{n+1} &= \beta_k|\beta_n = \beta_j), \\ &= \sum_x p(\beta_{n+1} = \beta_k|x)p(x|\beta_n = \beta_j), \\ &= P_{jk}. \end{aligned}$$

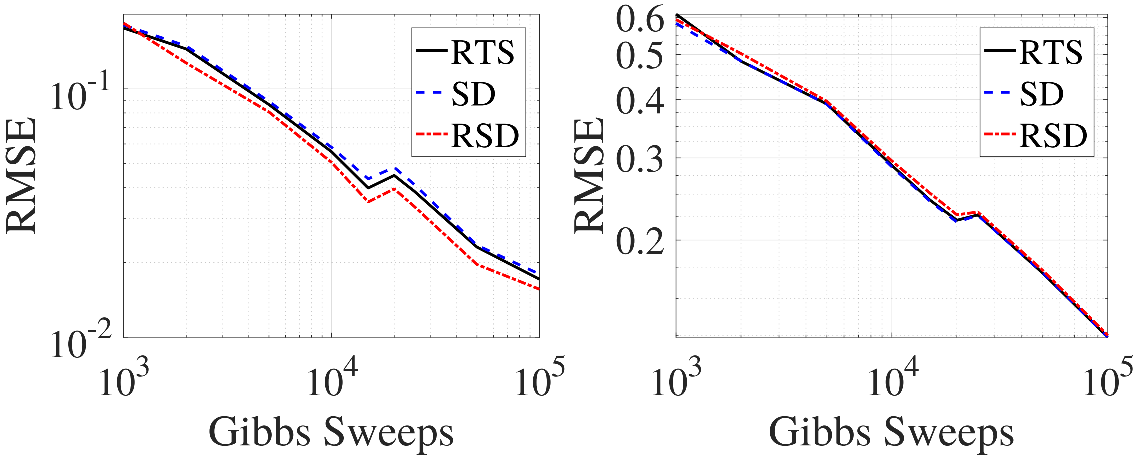

Figure 8. An illustration of the effect of estimating the stationary distribution from the transition matrix. Both plots show the RMSE on RBMs averaged over 20 repeats. Experimental procedure is the same as the main text. (Left) RTS, TM, and RTM compared on a 784-10 RBM. Because the latent dimensionality is small, mixing is very effective and accounting for the transition matrix improves performance consistently by about 10%. (Right) For an 784-200 RBM, the approximation as a Markov transition is inaccurate, and we observe no performance improvements.

Note that in general that we do not have an exact Gibbs sampler on $p(x|\beta)$. In these cases the approach is approximate. The top eigenvector of P gives the stationary distribution over β_k, which is $q(\beta_k)$. We briefly mention two importance sampling strategies to estimate this transition matrix. First, this matrix can simply be estimated with empirical samples, with

$$P_{jk} \propto \sum 1_{\{\beta_n+1=\beta_k,\beta_n=\beta_j\}},$$

where $1_{\{\cdot\}}$ is the identity function. Then $q(\beta_k)$ is estimated from the top eigenvector. We denote this strategy Stationary Distribution (SD). A second approach is to Rao-Blackwellize over the samples, where

$$P_{jk} \propto \sum p(\beta_n + 1 = \beta_k|x_n) 1_{\{\beta_n=\beta_j\}}.$$

We denote this strategy as Rao-Blackwellized Stationary Distribution (RSD).

The major drawback of this approach is that it is rare to have exact Gibbs samples over $p(x|\beta)$, but instead we have a transition operation $T(x_n|\beta, x_{n-1})$. In this case, it is unclear whether this approach is useful. We note that in simple cases, such as a RBM with 10 hidden nodes, RSD can sizably reduce the RMSE over RTS, as shown in Figure 8(Left). However, in more complicated cases when the assumption that we have a Gibbs sampler over $p(x|\beta)$ breaks down, there is essentially no change between RTS and RSD, as shown in a 200 hidden node RBM in Figure 8 (Right). Our efforts to correct the transition matrix for the transition operator instead of a Gibbs sampler did not yield performance improvements.