Response Letter(s)

|  |  |
| --- | --- |
| **Response Letter** | |
| **Response:** | We thank the reviewers for their comments and analysis.    \* R1:  1. Some setup details were omitted due to space restrictions. If our paper is accepted for publication, in the camera-ready version, we would like to omit an experiment and use the space to explain the setup of other experiments in detail.     \* R2:  \*Regarding the details of experiments see R1.1.   Regarding the details of the main algorithm if you refer to INVERSETRANSFORMSAMPLE, it is based on binary search and can be added to the camera-ready version. If details of integration and other symbolic operations are meant, the readers can refer to the open source software that we will release by publication.   \*We only referred to BUGS since it is famous otherwise no probabilistic programming framework so far has been able to handle non-linear deterministic constraints. See R3.1&3.2 for more discussions.   3. Yes, U(\*,\*) is uniform distribution.   4. Error thresholds have been 3.0 for all experiments. As far as this threshold is not too low (so that some algorithms may not reach it at all) or too high (so that algorithms reach it instantly), the comparative results are similar.   5. How the exact answer computed in experiments 3&4:  As mentioned in page 6, the exact answer for experiment 3 is 0 since the difference of two parallel MCs is measured. For experiment 4, the exact answer is computed manually using the fact that the problem is symmetric.   7. To realize SOLVE method, we have implemented our own symbolic equation solver.   8. By converting hard constraints to soft penalty distribution, the problem of observation of deterministic random variable would be transformed to (i.e. approximated with) the problem of observation of stochastic random variables that is already addressed in the literature.     \*R3:   1. Toolboxes that support sampling with deterministic variables:  Up to our knowledge, no existing probabilistic programing toolbox has handled deterministic (continuous) OBSERVED random variables so far. Clearly, all full-fledged such toolkits do support deterministic functions/nodes/random variables. But these random variables cannot be observed. We are not experts of Stan but we did not find any clue in its manual indicating that it handles deterministic observed variables and due to the problems that such observations create for functions which are beyond the scope of particular algebraic forms, it is really unlikely that Stan can handle such dependencies.  We know that the solution sought be Anglican toolbox is to approximate deterministic observed random variables by low variance observed variables. For example in introductory example “addition” of the (Anglican toolbox website), observation “a+b=7” has to be approximated by observation “z=7” where z is an auxiliary normal random variable centered at (a+b) and variance 0.00001. By decreasing the variance, the accuracy increases but sampling performance decreases and sampling becomes impossible when variance tends to 0.  Our algorithm works because we reduce the dimension while other existing algorithms do not. Thus, we insist that the novelty of our work is to handle deterministic continuous observed random variables for the first time.   2. Note than in PyMC, deterministic variables have no observed flag. Therefore, like the other existing frameworks, PyMC cannot model deterministic observed variables.   3. In the paper https://www.cs.berkeley.edu/~russell/papers/aistats13-dysc.pdf all relevant theorems only hold for a very simple Bayesian network with a single parameter vector, i.i.d. data and summation as the deterministic constraint. They claim that their work can be generalized to arbitrary Bayesian networks and “other” deterministic constraints. But by following their theorems we did not see an obvious generalization mechanism. It also seems that for each particular form of constrain, a distinct solution should be found (i.e. reducing multiplication to summation using logarithms and unspecified solutions for more complicated constraints). Even for a simple addition constraint they claim that the problem is NP-hard however their proof does not seem to hold for the continuous case and it is obvious that linear deterministic constraints are already handled effectively. For these and other concerns we had eventually decided not even to cite this paper in our submission.   4. The complexity order of symbolic integration is linear in the number of integrand terms and it is not a bottleneck. Collapsing a determinism in form Z=A/(B\_1 … B\_n) can grow the model by 2^n (where B\_i are terms with relative degree less than 4) but this is only problematic for very complicated constraints.   5. The proposal distribution for the rejection sampler has been uniform in a hypercube surrounding the prior. We accept that more complicated proposals could serve experiment 3.   6. InverseTransformSample is implemented by Binary search.   7. Inverse CDF can be computed symbolically leading to a completely analytical Gibbs. However, its general implementation is very hard and since binary search is quite effective, it has not been in our agenda so far. |
| **Time:** | Oct 26, 23:31 GMT |
| **Letter:** | NOTICE: Due to the impossibility of opening the review response period  at 5pm PDT, October 22, we are extending the period until October 26,  7pm PDT. The new period overlaps the weekend but it has been extended  significantly from 2 days to more than 3 days.    NOTICE: In order to start discussion as soon as possible, we ask you  to enter your response as soon as it is finished instead of waiting  until the deadline. Likewise, if you don't want to enter a response,  we ask you to enter an "empty response". In this way, we can kick  off discussions as papers become "ready".     Dear [\*FIRST-NAME\*],   Thank you for your submission to AAAI-15. The AAAI-15 review response  period is NOW through 7pm PDT on Sunday, October 26.   During this time, you will have access to the current state of your  reviews and have the opportunity to submit a response of up to 800  words via Easychair. Please keep in mind the following during this  process:   \* The response is intended as an opportunity to correct any factual  errors in the reviews and to answer any questions posed in the  reviews. Please do not provide new research results or reformulate  the presentation. Please also try to refrain from rebutting  subjective opinions. Try to be as concise and to the point as  possible.   \* The review response period is an opportunity to react to the  reviews, but not a requirement to do so. Thus, you do not need not  respond if you feel that the reviews are accurate and do not pose  any questions. However, please read the second notice above.   \* The reviews are as submitted by the PC members, without any  coordination among them. Thus, there may be inconsistencies among  the reviews. While we have tried hard to get 3 reviews per  submission by this point, it is inevitable that some reviews are  missing given that AAAI-15 received about 2,000 submissions.  However, we can confirm that every paper has at least 2 reviews,  the vast majority (> 98%) have 3 reviews, and some papers have 4  or more reviews.   \* Where necessary, we are working on getting additional reviews in and,  thus, an additional review could show up during the author response  period. After the author response period, the reviews may be updated  to take into account the discussions among the reviewers and  additional reviews may be solicited.   \* The program committee will read your response carefully and take  this information into account during the discussion, but they may  not directly respond to your response in their reviews.   \* Your response will be seen by all PC members who have access to the  discussion of your paper, so please try to be polite and  constructive. If\_you\_abuse\_the\_word\_limit, your response might get  deleted.   \* No edits or deletion of comments are possible after the response has  been submitted on EasyChair. Only the corresponding author(s) are  able to enter a response.   The reviews on your submission are attached to this letter. To submit  your response you should log on the EasyChair Web site  (https://www.easychair.org/conferences/?conf=aaai15) and select your  submission. The author response period ends promptly at 7pm PDT on  Sunday, October 26, and no author feedback will be allowed after that  time.   Thank you,   Blai Bonet and Sven Koenig  AAAI-15 Program Cochairs     [\*REVIEWS\*] |
| **Time:** | Oct 23, 13:15 GMT |

Reviews

|  |  |
| --- | --- |
| **Review 1** | |
| Significance of the Contribution: | 8: (+++) |
| Soundness and Positioning with Respect to Related Work: | 8: (+++) |
| Depth of Theoretical and/or Experimental Analysis (as appropriate): | 7: (++) |
| Quality of Presentation: | 8: (+++) |
| SUMMARY RATING: | **4**: (++++) |
| Comments for the Authors: | This paper addresses the problem of handling deterministic constraints between random variables in approximate inference strategies like Gibbs sampling. The main idea of the symbolic Gibbs sampling method introduced is to represent possibly nonlinear deterministic rules as polynomial fractions, and to overcome computational problems by analytically pre-computing the univariate cdfs required in Gibbs sampling. In my opinion this work addresses a problem that is both relevant and challenging, where relevance stems from the fact that deterministic relations between random variables are common in many real-world inference scenarios, and challenge refers to the inherent difficulties that arise in traditional MCMC methods with mixed stochastic-deterministic relations.  The paper is written in a clear and transparent way. Both the introductory and the technical parts are easy to follow. The only (minor) problem I see is the not fully convincing experimental validation: the experiments provided are OK, but for some experiments (particularly exp 3 and 4) I found it non-trivial to follow the exact experimental setup and to estimate the resulting difficulty of the task. |

|  |  |
| --- | --- |
| **Review 2** | |
| Significance of the Contribution: | 6: (+ (slightly positive)) |
| Soundness and Positioning with Respect to Related Work: | 6: (+ (slightly positive)) |
| Depth of Theoretical and/or Experimental Analysis (as appropriate): | 5: (- (slightly negative)) |
| Quality of Presentation: | 4: (--) |
| SUMMARY RATING: | **-1**: (- (slightly negative)) |
| Comments for the Authors: | The paper presents a new inference scheme called Symbolic Gibbs sampling for hybrid graphical models that have non-linear deterministic constraints. It introduces a class of problems (Polynomial-Piecewise Polynomial Fractions) that facilitates inference in the presence of linear/nonlinear algebraic deterministic constraints.   In my opinion, the most interesting idea in the paper (and I believe is the main contribution of the paper) is marginalizing a variable by symbolically integrating (indefinite integral) it and then evaluating the integral in presence of the constraints.   However, although the idea is interesting, the paper fails to provide important details about the experiments as well as the main algorithm. Ihe authors have failed to formally define many notations used. It seems like the main inspiration behind this work is limitation of the BUGS system (I am not sure how often in practice non-linear constraints appear).  Minor points and questions:  1> In step 3, Algorithm 1 (Joint factor formulation): The algorithm requires to multiply over all the factors and create a joint factor. This step can itself be intractable since the size of the joint factor will be exponential in the number of variables. An alternate (better?) way might be to use the Shenoy-Shafer architecture. 2> The Dirac delta has been used after equation (1) without mentioning that delta is Dirac delta. 3> The U(\*, \*) distribution is not defined. Is it a uniform distribution (I am assuming it is)?  4> What are the error threshold measures used in Experiment 3 and 4? 5> How the error is computed for Experiment 3, and 4? (i.e.- how the exact answer is computed?) 7> In the step 4 of Algorithm 1 how is the SOLVE method implemented (did you use any symbolic equation solver, or is it hand computed for the example problems?)  I am curious to know how the sampling algorithm performs if these hard constraints are converted to some form of penalty distribution (which heavily penalizes samples that violate the constraint). For example instead of using (hard) Dirac function, may be a softer penalty function (like a gaussian penalty, e.g: N(-(x-G^x)^2, \epsilon) for constraint x=G^x) could have been used. This might have helped the collapsing steps. |

|  |  |
| --- | --- |
| **Review 3** | |
| Significance of the Contribution: | 7: (++) |
| Soundness and Positioning with Respect to Related Work: | 3: (---) |
| Depth of Theoretical and/or Experimental Analysis (as appropriate): | 5: (- (slightly negative)) |
| Quality of Presentation: | 7: (++) |
| SUMMARY RATING: | **1**: (+ (slightly positive)) |
| Comments for the Authors: | This paper seeks to address the challenge of doing MCMC in graphical models that have deterministic constraints amongst the variables. The key idea in the paper is to define a class of models such that the following algorithm will work to compute a single-site Gibbs sampling update for variable i: 0. Replace all determistically-constructed variables in the model with the expressions of other variables that they are constrained to be equal to. 1. For each remaining variable, symbolically express the conditional pdf P(xi | x\_{-i}).  Solve for xi in terms of other variables. 2. Symbolically integrate the result to get a symbolic conditional CDF for xi. 3. Use CDF inversion (presumably using binary search) to sample xi | x\_{-i}.  The paper then suggests to do steps 0-2 as a preprocessing step, so that only step 3 is required for each Gibbs update. Experiments show this yields around an order of magnitude speedup over a Gibbs sampler that does steps 1 & 2 nonsymbolically but at every iteration of the sampling.  There are some pros and cons to this paper. The main pro is that the approach is reasonable, and the high level idea of doing some symbolic pre-processing to speed up MCMC inference is compelling. The main con is that there is lots of related work that is ignored in the text and not compared to experimentally.   Pros:   - The paper appears technically sound, is well organized, and the motivation is clear.  - The class of models, Piecewise Algebraic Graphical Models, that are  presented along with the sampling method seems like an interesting and powerful class of model.  - The use of symbolic computations to speed up Gibbs sampling for a general class of models is interesting (of course, people already apply this sort of reasoning when building one-off Gibbs samplers for individual problems, when they construct and simplify conditionals with pen-and-paper, then implement the simplified updates).   Cons:   - The treatment of related work is quite simplistic. More advanced methods of MCMC are quite common these days and have made their way into general-purpose toolboxes, e.g., HMC in Stan [B], SMC methods in Anglican [C], and these toolboxes support sampling with deterministic variables. Thus, the discussion in the intro is over-stating the novelty of the paper. It may be the case that the current paper can support models where these approaches would fail, or that it works better, but this discussion is more nuanced than what appears in the paper, and it should be included. Also related is [A] and the citations within.  - Related point: the baselines are too weak to convince me that this is the best approach. Stan and Anglican are readily available to download and should be compared against when possible. At a quick glance, PyMC [3] appears to support at least the Collision example using the Deterministic class within the modelling language.  - There needs to be a precise discussion of the runtimes of the symbolic computations. Please give expressions in Big-O notation of the solving of the symbolic equations and of the symbolic integration. Is there any case in which these costs will grow exponentially? Relatedly, the collapsing determinism can presumably lead to exponential blow-up in the size of the model representation. This should be discussed as well.  - All of the experiments are toy.   In total:  I found the content of the paper to be interesting, and I think with a bit of revision and more nuanced discussion of related work and stronger baselines, this would be a good paper.    Detailed points / questions:  - Please move the legend in fig 4 (c)  - What is the proposal distribution for the rejection samplers?   - I didn't understand the discussion of the inapplicability of rejection sampling in experiment 3. Just because the log pdf is unbounded doesn't mean there isn't some proposal distribution that could serve as an appropriate envelope.  - Are there situations where it is possible to go one step further and symbolically compute an inverse CDF function, so that step 3 above would not require binary search?  - "... source voltage V and observed" -> "... source voltage V are observed" ->   - How is InverseTransformSample implemented? Binary search to invert the CDF?   [A] https://www.cs.berkeley.edu/~russell/papers/aistats13-dysc.pdf [B] http://mc-stan.org/manual.html [C] http://www.robots.ox.ac.uk/~fwood/anglican/ [D] http://pymc-devs.github.io/pymc/ |