IJCAI-16 (author)

IJCAI-16

Help Log out

New Submission | Submission 2164 | IJCAI-16 | Support | News | EasyChair

Author Response Information for Submission 2164

Response Letter(s)

Response Letter	
Response:	We thank all the reviewers for the helpful comments. We now address the main questions raised. R1: “We could include strings of various lengths in the basis of standard PSR, why is that not enough to capture the multiple timing scale structure” This only helps to *discover* states which need longer sequences of actions in order to be reached, but doesn’t improve the learning of frequent multi-step transitions. “My guess is that PSR predictions suffer from the issue of errors cumulating over multiple matrix multiplications, but I am not sure.” This is indeed part of the reason why M-PSRs work, but the issue gets deeper when one looks at models with a reduced number of states. In that case, M-PSRs show that frequent multi-step transitions can be more accurately represented by a specialised multi-step operator than by a product of generic single-step operators. When the model gets larger this advantage of the specialized operators disappears, but the update time increases. Thus, M-PSRs can improve the prediction accuracy on frequently occurring events and at the same time help control the complexity of update operations. In future work we plan a formal theoretical analysis of the trade-offs between these two aspects of M-PSRs. As suggested, we will include an example of the case when M-PSRs help in the paper. We will also address the minor issues pointed out in the revision. R2: Regarding the impact of the choice of P and S, as in the standard PSR, the quality of the learned M-PSR is affected by the quality of the basis used. In our experiments we choose the basis as explained at the end of page 4 and beginning of page 5. The construction is informed by prior work and our own experimental work with the presented domains. “Is the greedy scheme optimal in any sense?" It’s not clear whether the presented scheme is optimal except in some special cases. It could indeed by substituted by a different compression scheme, as long as the compression of a sequence can be efficiently recomputed every time a new symbol is observed. We will include running time information for the experiments in the revised version of the paper as well as time for computing the answers to queries, in the revised version. R3: "the state representation at time n may correspond to a completely different sequence of updates than that at time n+1 [...] you could apply the traditional update to state n, and compare that to the M-PSR state update. This would give an idea of variability” This is indeed an interesting question we will explore in future

	work. Nonetheless, we want to point out that the update at time n+1 is not a “completely different sequence of updates”: it will be the same sequence of updates up to a change limited by the length of the longest multi-step operator, as explained at the beginning of page 4. We are working on a theoretical analysis but it goes beyond the scope of this paper.
Time:	Mar 12, 04:13 GMT
Letter:	Dear [*NAME*] Thank you for your submission to IJCAI-16. Author response period has now started. Your responses will be accepted between now and 11th March 11:59PM UTC-12 (that is as long as it is 11th March anywhere in the world). During this time, you will be able to read the current reviews for your paper and have the option to submit a response of up to 600 words (that is a whole hundred words more than those "other" AI conferences!). Please keep the following points in mind: * Only a corresponding author is able to enter the response. * Submitting an author response is optional; it is not a requirement. * An effective response will identify any factual errors in the reviews and focus on any questions posed by the reviewers. It is normally ineffective to attempt to provide new research results, dispute matters of judgment, or reformulate the presentation. Try to be as concise and to the point as possible. * The reviews are provided as submitted by the PC members, without any coordination between them. Thus, there may be inconsistencies. Furthermore, these are not the final versions of the reviews, since they will be updated to take into account your feedback and any discussions among PC members. It might also be necessary to solicit additional reviews after the author response period has closed. * The program committee will take author responses into account during the discussion period. We will strongly encourage reviewers to update their reviews to reflect the discussion and react to your response; however, we cannot guarantee that all reviews will be so updated.

	* No edits or deletion of comments are possible after the response has been submitted on EasyChair. The reviews on your paper are attached to this letter. To submit your response you should log on the EasyChair Web site https://easychair.org/conferences/?conf=ijcai16 and select your submission on the menu. Best Wishes Rao ps: Go AlphaGo o-|< --- Subbarao Kambhampati IJCAI-16 Program Chair http://rakaposhi.eas.asu.edu ======REVIEWS OF YOUR PAPER FOLLOW==== [*REVIEWS*]
Time:	Mar 10, 00:19 GMT

Reviews

Review 1

Significance:	3: (high (substantial contribution or strong impact))
Soundness:	3: (correct)
Scholarship:	2: (relevant literature cited but could expand)
Clarity:	3: (well written)
Breadth of Interest:	3: (some interest beyond specialty area)
Summary Rating:	4: (++++)
Summarize the Main Contribution of the Paper:	The paper proposed Multi-step Predictive State Presentation (M-PSR), an extension to the standard PSR model aiming at handle frequent patterns occuring at multiple time scales, which subsumes the previous timing model as its special case. This is analogues in spirit to the option framework in reinforcement learning, and similar gains are expected in the context of learning models of dynamical systems. A detailed model construction procedure is developed. Empirical evaluations demonstrate the effectiveness of the model compared to standard approaches, and the learned operators are found to capture the nature of the domains.

Comments for the Authors:	The paper is well motivated, clearly presented, and supported by detailed empirical results. Improving the practicality of PSR (esp. with spectral learning) has been of substantial interest in the community, and one of the reasons for PSR not working well in practice, as argued by the paper, is the lack of ability to incorporate domain structure, and this paper considers a particular structure of multiple scale of events. In general I like this paper. Still, I hope the author(s) could elaborate more on the following question to help me (and other readers) understand the strength and value of M-PSR: Why do we want to use M-PSR? Yes, you have experiments where M-PSR outperforms standard PSR. But can you provide a simple, sharp, and potentially theoretical explanation for that? We could include strings of various lengths in the basis of standard PSR, and why is that not enough to capture the multiple timing scale structure of a problem? (My guess is that PSR predictions suffer from the issue of errors cumulating over multiple matrix multiplications, but I am not sure.) I notice that you cited the option framework in RL as an analogy; regarding options, we do have a simple yet fundamental explanation on why it's useful: using options shrinks the discount factor. Can you provide something similar here, and maybe accompany by a minimal running example? Minor issues - The example on Base M-PSR in Sec 2.1: shouldn't b be such that b > 1? - Figures are too small (need something like 300% zoom to see on the screen, and no way to see when printed out). - Do standard PSR and M-PSR share the same basis in the experiment? - It would be interesting to see experiment results on natural language domains. For example, in character-level language modelling, would M-PSR learn something like common word prefix/suffix/root?

Review 2

Significance:	**2**: (medium (modest contribution or average impact))
Soundness:	**3**: (correct)
Scholarship:	**3**: (excellent coverage of related work)
Clarity:	**3**: (well written)
Breadth of Interest:	**2**: (limited to specialty area)
Summary Rating:	**3**: (+++)
Summarize the Main Contribution of the Paper:	This work presents the multi-step predictive state representation (M-PSR) which augments the canonical PSR with a set of multi-step observations, a coding function, and a modified set of transition matrices. The authors describe a spectral learning algorithm for learning data-driven M-PSRs using Hankel matrices. Experimental results on two simulation environments

	(Double Loop and Pacman) show that the M-PSR outperforms the PSR in terms of norm difference between the estimated and true probability distributions. Overall, the paper is well-written and describes an interesting extension to PSRs. Sections 1 and 2 provide a good background and motivate the problem well. Although the experiments are limited to two "toy" problems (plus one modification), they serve to illustrate the utility of the representation (particularly for small model sizes), and that the learning algorithm manages to identify the "correct" operators from the training data. A few comments I think will help readability: - Sec 3.1: A brief statement on choosing P and S would be helpful (and make the paper more self-contained). From my reading, the choice of these sets can have a significant impact on the representation? A natural question is how sensitive the representation is to "poor" basis choices? - Sec 3.4: Is the greedy scheme optimal in any sense? More generally, can any suitable compression method be substituted in? - Sec 4: I couldn't find information related to how long it took to learn the M-PSRs, relative to the typical PSR. A concern is the scalability of this approach to large-scale problems involving very large state and observation spaces. Can the authors comment? - Sec 4: Figure 5:, the panels on right (stretch factor) have different model size range from the left. Just to clarify, were the ranges truncated because the M-PSR and the PSR performed similarly beyond 40? - Sec 5: mentions briefly potential computational gains when performing conditional probability queries since M-PSR uses smaller matrices. This is certainly a nice feature of having a compressed representation; can the authors give some quantitative timing results of the difference in computational times in the domains studied? Minor comments: - Sec 3.3, 1st par, line 2, "observations"->"observation" - Sec 5, left col, last line, this sentence (which continues on to the right column) is grammatically incorrect. - Fig 6, caption, "(left"->"(left)" - I found the graphs a little hard to read. Given the page limit, I can empathize, but I think Fig 2 right (the Pacman graph) could be replotted to reduce height, which would give some additional space to increase the size of the very small figures.
Comments for the Authors:	

Review 3

Significance:	**2**: (medium (modest contribution or average impact))
Soundness:	**3**: (correct)
Scholarship:	**2**: (relevant literature cited but could expand)
Clarity:	**2**: (mostly readable with some scope of improvement)
Breadth of Interest:	**3**: (some interest beyond specialty area)
Summary Rating:	**2**: (++)
Confidence:	**3**: (highly confident)
Summarize the Main Contribution of the Paper:	This paper presents an interesting modification to spectral PSRs, by introducing the idea of multi-step update functions (matrices). This is particularly useful when developing system models that have state representation that is more compact than the true model (e.g. has many fewer entries in the state vector than are needed for a complete representation). The stated motivation for this is that the multi-step updates can implicitly contain updates in a compact form that capture the contributions of each step. This motivation, in fact, is the key advantage to using these multi-step PSRs. The paper develops the multi-step update algorithms and analyzes the algorithm on a few mid-sized problems. However there is no theoretical contribution, the methods are only verified empirically. The paper is generally clear and well-written, although it would benefit from clearly stating up front that the goal was to develop multi-step state updates, rather than multi-step predictions (which already exist in traditional PSR formulations). The empirical investigation certainly suggests that benefits are possible by using the M-PSR over PSRs, especially on compact representations. However, further investigation on a larger range of problems would be necessary before that statement could be broadly applied. One other area that is somewhat troubling is that a state update must be learned for each of the multi-step sequences. Therefore, this representation does increase model size, and thus also increase the amount of required training data (especially because the training data for a given multi-step sequence must correspond to that particular multi-step sequence occurring in the data).
Comments for the Authors:	There are two areas that would benefit from further theoretical investigation. First, an analysis of the connection between many single-step

updates and a single multi-step update. Some sort of approximation accuracy analysis for these compact updates would strengthen the paper considerably. For instance, a compact representation could be projected to a full representation, then many single-step updates applied, followed by projection back to the compact repn. This could be compared to the compact multi-step update.

Second, if I understand correctly, the state representation at time n may correspond to a completely different sequence of updates than that at time n+1. In other words, it is not simply a matter of updating state n to get state n+1. Understanding the consistency of state representation over time is also interesting. Because the single-step observations are in \Sigma', you could apply the traditional update to state n, and compare that to the M-PSR state update. This would give an idea of variability within this representation.