We thank the reviewers for their insightful comments.

Table 1: Comparison to other SOTA models. "R": RESCAL. "C": ComplEx.

	WN18RR			FB15k-237		
	MRR	H@1	H@10	MRR	H@1	H@10
RotatE	.476	.428	.571	.338	.241	.533
MuRP	.481	.440	.566	.335	.243	.518
HAKE	.497	.452	**.582**	.346	.250	.542
TuckER	.470	.443	.526	.358	.266	.544
R-Reg_p1	.281	.220	.394	.310	.228	.338
C-Reg_p1	.409	.393	.439	.316	.229	.487
R-DURA	**.498**	**.455**	.577	.368	**.276**	.550
C-DURA	.491	.449	.571	**.371**	**.276**	**.560**

Response to Reviewer 1

Comparison with SOTA distance-based methods We compare our models with some SOTA distance-based (DB) models, including RotatE [21], MuRP [Ref1], and HAKE [30]. Table 1 shows that, RESCAL-DURA and ComplEx-DURA perform competitively with the SOTA DB models. RESCAL-DURA outperforms all the aforementioned DB models in terms of MRR and H@1.

The optimal value of p In DS models, the commonly used p is either 1 or 2. When $p = 2$, DURA takes the form as the one in line 180. If $p = 1$, we cannot expand the squared score function of the associated DS models as in line 138. Thus, the induced regularizer takes the form of $\sum_{(h_i,r_j,t_k)\in\mathcal{S}} \|\mathbf{h}_i\bar{\mathbf{R}}_j - \mathbf{t}_k\|_1 + \|\mathbf{t}_k\mathbf{R}_j^\top - \mathbf{h}_i\|_1$. The above regularizer with $p = 1$ (Reg_p1) does not gives an upper bound on the tensor nuclear-2 norm as in Theorem 1. Moreover, experiments show that, DURA significantly outperforms Reg_p1 on WN18RR and FB15k-237 (see the third and fourth parts of Table 1). Therefore, we choose $p = 2$.

Analyses for non-diagonal relation matrices Tensor nuclear-2 norm is defined based on the CP decomposition [6]. When relation matrices are non-diagonal, TFB models do not take the form of the CP decomposition. Therefore, Theorem 1 does not apply to the non-diagonal case. Moreover, as stated in [6], computing the nuclear-2 norm of a 3-tensor over $\mathbb{R}$ is NP-hard, which implies that numerical analyses are intractable.

Other suggestions We will improve our paper accordingly.

Response to Reviewer 2

Confusing writing at times WN18RR used in our paper is the same as that in the original paper [22], of which the number of entities is $40,943$. Thanks for pointing out this typo. We will correct it accordingly. In line 193, A is a placeholder that can be any matrix. That is, for any matrix A, $A(i,j)$ represents the entry in the i-th row and the j-th column of it. We will polish it in the final submission, if accepted.

More recent KGC algorithms included in the evaluation Table 1 shows the evaluation results of our methods against recent DB models and TuckER [Ref2]. We will include the results in the final submission, if accepted.

Response to Reviewer 3

Tensor factorization-based (TFB) models in this paper is actually ComplEx TFB models in our paper (e.g., lines 130-131) can be CP, ComplEx, and RESCAL. Note that, both the real part and the conjugate of a real matrix are equal to the matrix itself. When all the embeddings are real, the score function in lines 130-131 corresponds to CP or RESCAL.

Why just not use the score function in line 134 instead As the regularization coefficient is usually small, a TFB model regularized by DURA is not the same as its associated DB model. The regularized model is dominant by the score function of the TFB model. Therefore, DURA does not aim to make models behave like the score function in line 134. Instead, it aims at introducing the prior knowledge that, tail (head) entities—connected to a head (tail) entity through the same relation—should have similar embeddings. Hence, we cannot just use the score function in line 134.

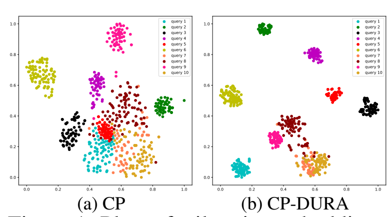
The statements in lines 160-161 are imprecise "TFB models" in these lines corresponds to TFB models without any regularization. Thus, it does not include models with N3. We will provide more details in the final version, if accepted.

RESCAL+N3 The definition of N3 regularization [13] depends on a summation of R vector norms, where R is the tensor rank (see Section 4.1 in [13]). Note that, tensor ranks used in [13] are defined based on the CP decomposition. As RESCAL does not take the form of the CP decomposition, we cannot apply N3 regularization to RESCAL.

Prior work on regularizing factorization models and KGE We will cite these papers in the final version, if accepted.

Response to Reviewer 4

T-SNE plot Suppose that (h_i, r_j) is a *query*, where h_i and r_j are head entities and relations, respectively. An entity t_k is an *answer* to a query (h_i, r_j) if (h_i, r_j, t_k) is valid. We randomly selected 10 queries in FB15k-237, each of which has more than 50 answers. Then, we use T-SNE to visualize the answers' embeddings generated by CP and CP-DURA. Figure 1 shows that, with DURA, entities with the same (h_i, r_j) contexts are indeed being assigned more similar representations. We will include the results in the final version, if accepted.



(a) CP (b) CP-DURA

Figure 1: Plots of tail entity embeddings.

Would DURA help on other KGC models such as KBAT/GAATs The answer is no. DURA is designed for tensor factorization based models and can bring significant improvements for them. However, it does not apply to models in other categories, such as KBAT/GAATs. We will discuss the potential limitation of DURA in the final submission, if this paper is accepted.

Some of the recent work on KGC has been omitted We will cite all these papers in the final submission, if accepted.

[Ref1] I. Balaževiá, C. Allen, and T. M. Hospedales. Multi-relational Poincaré Graph Embeddings. NeurIPS 2019.

[Ref2] I. Balaževiá, C. Allen, and T. M. Hospedales. TuckER: Tensor factorization for knowledge graph completion. EMNLP 2019.