Thank-you for the thoughtful and constructive feedback on our manuscript. Below, citation numbers refer to the main text and new references are made in-line.

We have now run experiments showing that autofocus can still be beneficial in *higher dimensional design spaces*. For the superconductor design experiment, originally conducted on the 10 most informative features, we have run the same experiment on 60 features (leaving out the 21 least informative features due to colinearity issues). As a point of comparison, many Bayesian optimization methods are considered practically useful up to around ten or so dimensions (Wang, Zoghi, *et al.*, IJCAI 2013). Presently, 60 is already a stretch goal for many problems. We find that autofocus maintains its statistically significant gains over non-autofocusing for the trust region EDA, CbAS (Table A1) at dimensionality of 60. We have not yet examined the other methods, but this result alone already demonstrates that there is no inherent problem to using autofocus in higher dimensions. We will pursue more systematically investigating the effect of dimensionality, across all methods.

Table A1: As Table 1 in the main text, but with 60 dimensions. (*) means p-value < 0.05, (**) means $p < 0.01$

	Median	Max	ρ	RMS
		CbAS		
Original	44.7	96.0	-0.12	26.4
Autofocused	52.0	103.4	0.12	21.6
Mean Diff.	**7.2****	**7.3****	**0.23****	**-4.7****

If one defines *extrapolation* as going into regions of the input space where the property (*e.g.*, $\mathbb{E}[y \mid \mathbf{x}]$) takes on very different values from the training labels, then indeed, this does occur in our experiments. In particular, all autofocused methods except Random Search always produced *top candidates with ground-truth expectations greater than the maximum training label*. We have now computed the probabilities that the methods produce top candidates with ground-truth expectations that surpass the maximum training label (a "probability of improvement"). For example, non-autofocused CbAS yields a probability of improvement that is 28 times higher than that of a random control (*i.e.*, drawing the same number of samples from the training distribution) ($p < 0.01$, Mann-Whitney U-test). Autofocused CbAS increases this to 65 times ($p < 0.01$). As for whether we are interpolating within the training data, we are not in the sense that the effective sample size (ESS) reported in the original submission shows that the search model deviated quite considerably from the training distribution (Fig. S3), as the ESS is the inverse of the exponentiated Renyí-2 divergence. Similarly, in additional experiments, we now find that the KL divergence between the training and test distributions goes up to 150 for DbAS and up to 10 for CbAS. All that being said, the goal of AF is a subtle one. We do not claim that AF *necessarily* makes a design method move further beyond the training data; instead, AF helps us make the best use of the oracle for wherever a design method does move, regardless of whether that movement is interpolation or extrapolation (also see "Oracle bias-variance trade-off" paragraph on page 4). From that point of view, using a method with a trust region complements, rather than contradicts, the goal of AF, although a more formal examination of how trust-region methods like CbAS can still enable extrapolation would be useful.

With respect to "reporting the expected probability of belonging to S, when say y_τ is the 80th percentile", if we understand correctly, this is tantamount to asking about extrapolation, and hence addressed in the previous paragraph. Note too that for maximization design problems, the desired S will be the set of values y such that $y \geq y_{\text{max}}$ (where $y_{\text{max}} \equiv \max_{\mathbf{x}} \mathbb{E}_{p(y|\mathbf{x})}[y]$).

With respect to *discrete design spaces*, in preliminary experiments on the protein design problem in (9), autofocused CbAS compared to CbAS yielded candidates with a maximum ground truth of 3.39 compared to 3.37 ($p = 0.03$) and a Spearman's ρ of 0.83 instead of 0.70 ($p = 0.01$). This topic requires more comprehensive experimentation which we continue to do.

The practical importance of the *fixed-data setting* has been demonstrated by recent protein engineering and materials design work, in which a single labeled dataset is collected, and a regression model is trained and used to guide the design of novel proteins without additional data (5, 6). Moreover, a prominent protein engineer has told us the fixed-data setting is what they are currently working on (Frances Arnold, personal communication). Finally, no matter how many rounds of data are acquired, at some point, the data are fixed and our current problem formulation applies.

With respect to the comment about non-negligible $P_0(y \in S)$, indeed, if this is too small, then the variance explodes. In effect, this says that the starting (training) data must have at least a hint of the property we care about, just as is required for successful directed evolution for protein design (3). We will add further clarification in the manuscript.

We will incorporate feedback on clarity of Fig 1, and also on our related work section, in particular linking to out-of-distribution generalization, small-area estimation, model misspecification, model inversion networks, and the Angermueller paper *etc*. As for formulating autofocus as a game, this was simply done to give a precise name to the problem formulation that emerged from our derivation. And indeed, Nash equilibria are hard, but as you note, empirically, we still find benefit from our approach. Finally, we apologize if some points were not addressed, but in light of time and space constraints we had to prioritize. We will of course address all points raised.