

# **Supporting Information: Design of polymer blend electrolytes through a machine learning approach**

Bill K. Wheatle, Erick F. Fuentes, Nathaniel A. Lynd, and Venkat Ganesan\*

*McKetta Department of Chemical Engineering, The University of Texas at Austin, Austin,  
TX USA*

E-mail: [venkat@che.utexas.edu](mailto:venkat@che.utexas.edu)

# Core Simulations

Table S1: Core simulation design parameters

ID	$\mu_{lo} (\mu_{EO})$	$\mu_{hi} (\mu_{EO})$	$N_A$	$N_B$	$\varepsilon_{AB} (k_B T)$	$\phi_A$	$\gamma$
1	1.679	1.798	5	33	0.975	0.950	1
2	1.728	2.230	14	36	0.518	0.104	1
3	1.711	2.038	21	29	0.632	0.741	1
4	1.345	1.546	19	34	0.878	0.106	0
5	1.522	1.996	20	32	0.544	0.020	1
6	1.607	2.064	8	25	1.214	0.528	0
7	1.650	2.026	4	36	1.015	0.440	1
8	1.680	2.110	27	40	0.770	0.476	0
9	1.524	2.115	23	40	0.805	0.454	1
10	1.582	2.133	22	29	0.540	0.155	0
11	1.720	2.235	28	31	1.096	0.844	0
12	1.297	1.864	23	39	1.119	0.922	0
13	1.662	2.123	1	34	1.158	0.762	1
14	1.362	2.249	19	22	0.637	0.189	1
15	1.689	1.809	16	27	1.003	0.392	0
16	1.517	1.759	13	38	1.076	0.098	1
17	1.684	2.234	3	37	1.124	0.889	0
18	1.302	1.950	4	29	0.896	0.668	0
19	1.700	1.999	12	39	0.606	0.249	0
20	1.283	1.445	4	15	1.034	0.179	0
21	1.429	2.196	19	39	0.927	0.106	1
22	1.603	1.812	2	37	1.140	0.779	0
23	1.531	1.884	23	32	0.594	0.599	1
24	1.642	2.108	6	18	1.196	0.371	1
25	1.749	2.200	29	34	0.813	0.263	0
26	1.508	1.687	26	29	0.751	0.694	0
27	1.602	2.244	21	28	1.071	0.453	1
28	1.454	1.993	2	38	0.538	0.035	0
29	1.413	1.730	17	28	0.804	0.483	1

As noted in the main text of this work, we simulated 29 core simulations as a training set with random design parameters. We present those parameters in Table S1. Each simulation

contained approximately 2000 polymer beads and enough cations and anions to maintain electroneutrality and a monomer/cation ratio of 16.

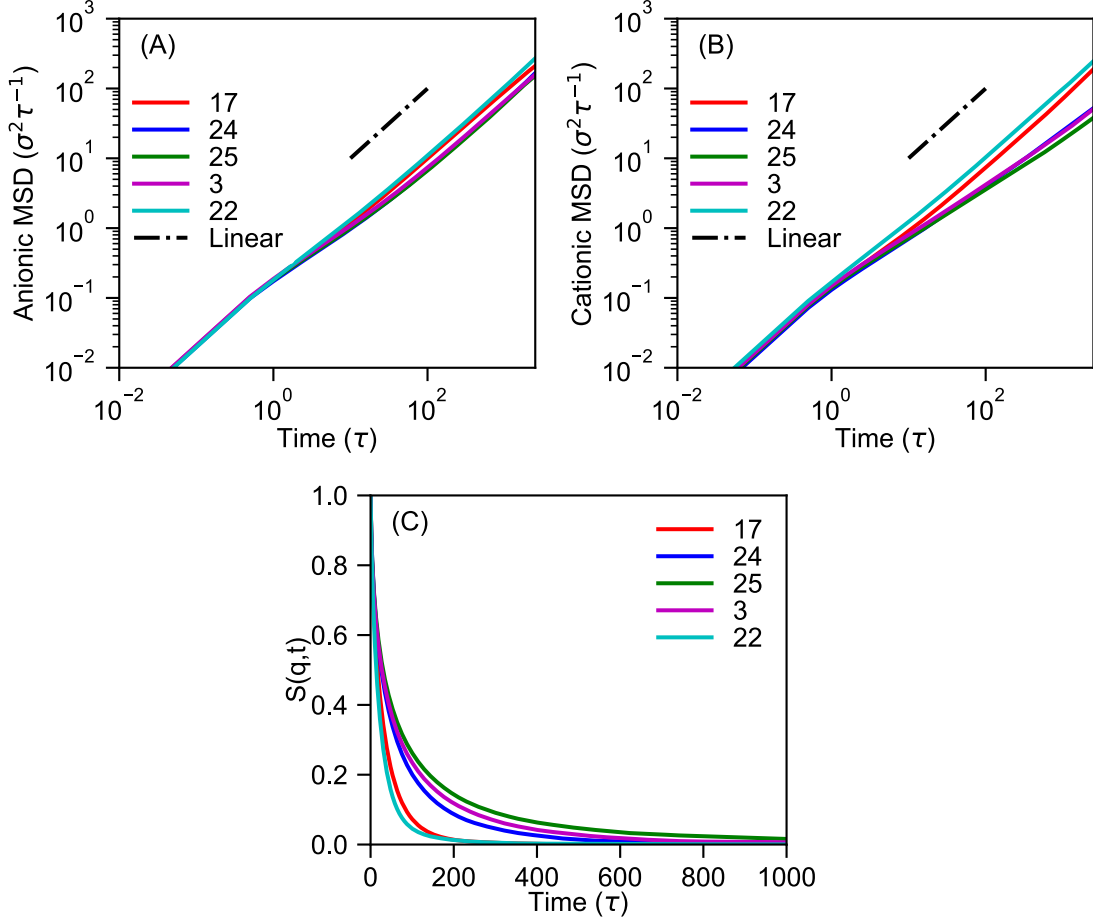


Figure S1: (A) Anionic and (B) cationic mean-squared displacements (MSDs) as functions of time. (C) The dynamic structure factor  $S(q,t)$  as a function of time. Each curve corresponds to a core simulation, whose number is specified in the legend. In panels A and B, the black dot-dashed line corresponds to a line. Most MSDs and all  $S(q,t)$  converge in the short time frame sampled.

In Figure S1, we plot the mean-squared displacements (MSDs) and dynamic structure factors  $S(q,t)$  of a sample of randomly chosen core simulations. It can be seen that all anionic MSDs (Figure S1A) and cationic MSDs (Figure S1B) reach linearity (or nearly so), shown by the black dot-dashed line. In addition,  $S(q,t)$  decays to 0 quickly for all core simulations

shown. For both measures, they approach convergence within the short sampling time span, which initially suggested that they may serve as proxy measures for conductivity and viscosity, as discussed in the main text.

## Bayesian Optimization

### Gaussian Process Regression

Bayesian optimization uses an underlying statistical model to iteratively optimize an expensive, black-box objective function  $f(\mathbf{x})$ , where  $\mathbf{x}$  is a vector of the input parameters for the objective function. Our chosen function and input parameter vector are described in the main text of this work. The prior distribution of  $f(\mathbf{x})$  is assumed to be described as a Gaussian process:

$$f(\mathbf{x}) \sim \mathcal{N}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \quad (\text{S1})$$

where  $\mathcal{N}$  is a normal distribution with mean function  $\mu(\mathbf{x})$  describing the expected value of  $f_p(\mathbf{x})$  and  $k(\mathbf{x}, \mathbf{x}')$  is the kernel describing the covariance of  $f_p(\mathbf{x})$  between points in the parameter space  $\mathbf{x}$  and  $\mathbf{x}'$ . Without significant loss, the mean function for all optimizations is assumed to be 0. The covariance function chosen for this work was the Matérn 5/2 covariance function:

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \left( 1 + \sqrt{5}r + \frac{5}{3}r^2 \right) \exp \left( -\sqrt{5}r \right) \quad (\text{S2})$$

where  $\sigma^2$  is the Matérn scaling parameter. The Matérn 5/2 covariance function was chosen based on the assumption that the kernel was twice-differentiable on the domain of  $\mathbf{x}$ .  $r$  is the distance between  $\mathbf{x}$  and  $\mathbf{x}'$  weighted by their corresponding length scales:

$$r^2 = (\mathbf{x} - \mathbf{x}')^\top \text{diag}(\boldsymbol{\ell})^{-2} (\mathbf{x} - \mathbf{x}') \quad (\text{S3})$$

where  $\text{diag}(\boldsymbol{\ell})^{-2}$  is a diagonal matrix containing length scales that set the relative importance of each parameter in  $\mathbf{x}$ .<sup>1</sup>

To ensure that the model learns from training sets to act as a predictive posterior for the optimization procedure, the prior distribution must be conditioned, according to Bayes' rule. Let  $\mathbf{f} = (f(\mathbf{x}_1), \dots, f(\mathbf{x}_n))$  and  $\mathbf{f}^* = (f(\mathbf{x}_1^*), \dots, f(\mathbf{x}_m^*))$  be vectors of training and test sets, respectively, of function evaluations of  $f(\mathbf{x})$ . Let  $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)$  and  $X^* = (\mathbf{x}_1^*, \dots, \mathbf{x}_m^*)$  be matrices of the input parameters. Finally, let  $K(X, X) + \sigma_n^2 I$ ,  $K(X, X^*)$ ,  $K(X^*, X) = K(X, X^*)^\top$ , and  $K(X^*, X^*)$  be matrices containing the covariances between training and test sets. It is important to note the  $\sigma_n^2 I$  term, which is a diagonal matrix that incorporates measurement uncertainty into the covariance matrix of the training set. Such a term builds in an assumption of imperfect measurement of the objective function, slows the convergence of the Gaussian process via conditioning (see Equations S4–S7 below), and increases the likelihood of exploration during the optimization process (see S10 and subsequent discussion below). The posterior (conditional) probability distribution for the test set  $p(\mathbf{f}^*|\mathbf{f}, X, X^*)$  follows from application of Bayes rule:

$$p(\mathbf{f}^*|\mathbf{f}) = \frac{p(\mathbf{f}^*, \mathbf{f})}{p(\mathbf{f})} \quad (\text{S4})$$

where  $p(\mathbf{f}^*, \mathbf{f})$  is the joint probability distribution between  $\mathbf{f}^*$  and  $\mathbf{f}$  and  $p(\mathbf{f})$  is the marginal probability distribution for the training set  $\mathbf{f}$ . The joint probability distribution is a Gaussian process of the following form:

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N} \left( \mathbf{0}, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) \end{bmatrix} \right) \quad (\text{S5})$$

and the marginal probability distribution is a Gaussian process of the form:

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K(X, X) + \sigma_n^2 I) \quad (\text{S6})$$

It can be shown, using Equations S4–S6, that the posterior distribution is a normal distribution of the form:

$$\begin{aligned} \mathbf{f}^* | \mathbf{f} \sim \mathcal{N}(K(X^*, X)(K(X, X) + \sigma_n^2 I)^{-1} \mathbf{f}, \\ K(X^*, X^*) - K(X^*, X)(K(X, X) + \sigma_n^2 I)^{-1} K(X, X^*)) \end{aligned} \quad (\text{S7})$$

It can be seen that the new mean function  $\mu = K(X^*, X)(K(X, X) + \sigma_n^2 I)^{-1} \mathbf{f}$  and the new kernel function  $K = K(X^*, X^*) - K(X^*, X)(K(X, X) + \sigma_n^2 I)^{-1} K(X, X^*)$ .

## Acquisition Function

The optimization procedure relies on determining the input parameters  $\mathbf{x}^*$  to evaluate via simulation. The common means by which such a goal is achieved is through the use of an acquisition function  $a(\mathbf{x}^*)$ :

$$a(\mathbf{x}^*) = \int_{-\infty}^{\infty} u(\mathbf{x}^*) p(f^* | \mathbf{f}) df^* \quad (\text{S8})$$

If the goal is to minimize  $f^*(\mathbf{x}^*)$ , then  $u(\mathbf{x}^*)$  is called a loss function. In this work, we choose the expected improvement (EI) loss function:

$$u(\mathbf{x}^*) = \max(0, f' - f^*(\mathbf{x}^*)) \quad (\text{S9})$$

where  $f'$  is the minimum function evaluation in the optimization. Evaluating the Equation S8 using the EI loss function yields the EI acquisition function:

$$a(\mathbf{x}^*) = (f' - \mu(\mathbf{x}^*)) \Phi(f' | \mathbf{f}) + k(\mathbf{x}^*, \mathbf{x}^*) p(f' | \mathbf{f}) \quad (\text{S10})$$

where  $\Phi$  is the cumulative distribution function of the posterior,  $\mu$  is the mean function described at the end of the prior paragraph, and  $k$  is the kernel described in the prior section. Choice of the appropriate  $\mathbf{x}^*$  occurs at the maximum of  $a(\mathbf{x}^*)$ . As can be seen, the EI acquisition function consists of two terms. The first drives the reduction of the mean function  $\mu(\mathbf{x}^*)$ . The second term drives an increase in the variance  $k(\mathbf{x}^*, \mathbf{x}^*)$ . As a result, the optimization will naturally balance between evaluating the objective function at  $\mathbf{x}^*$  that further reduces the objective function itself (exploitation) or reduce its uncertainty (exploration). Advanced sampling of the acquisition function by the method of local penalization was used in this work.<sup>2</sup>

## Optimization Results

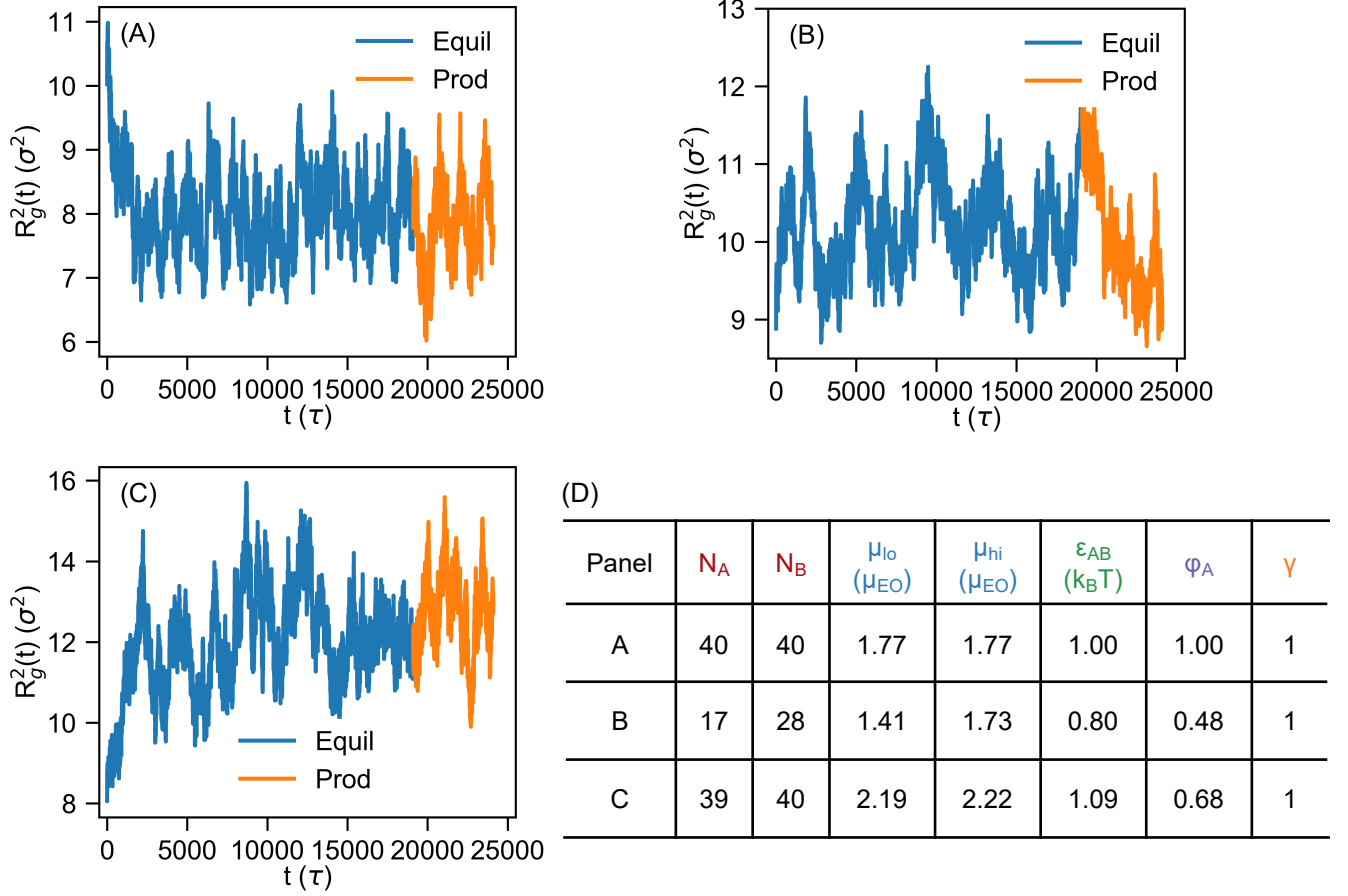


Figure S2: (A–C) Squared radius of gyration of a sample of runs of the optimization simulations for  $w_D = 0.5$  as functions of time. (D) Table of parameters for this sample. Equilibration step shown in blue and production step in orange.

As discussed in the main text, we calculated the radius of gyration  $R_g^2(t)$ . It can be seen that  $R_g^2(t)$  quickly tends to an equilibrium value (see panels A and C) or fluctuates about a constant value (see panel B), even for electrolytes that are slightly incompatible (panel C).



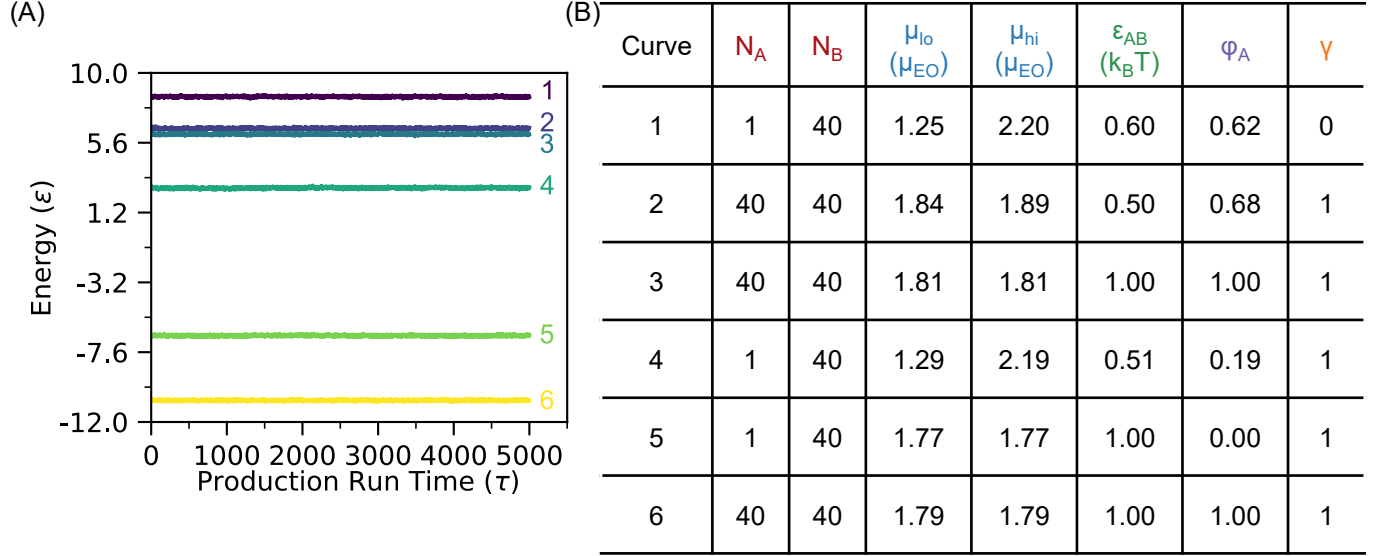


Figure S3: (A) Energies of random sample of production runs of the optimization simulations for  $w_D = 0.5$ . (B) Design parameters corresponding to each curve in (A). One can see a flat energy profile for each simulation.

As discussed in the main text, we calculated the energies of a sample of the production runs of the optimization simulations. It can be seen that the energy profiles are relatively flat, suggesting a steady state from which short-run proxy performance metrics can be reasonably calculated.

Table S2: Number of simulations run per  $w_D$  value

$w_D$	# Iterations	Iteration @ Max.	# Sims Run
0.0	15	10	45
0.1	15	6	45
0.3	15	9	45
0.5	20	5	60
0.6	31	20	93
0.7	28	19	84
0.9	15	6	45
1.0	20	12	60
Total (w/o core)	159	N/A	477
Total (w/ core)	160	N/A	506

As discussed in the main text, Table S2 summarizes the optimization process. It shows the number of iterations, the iteration at which  $f_p$  maximized, and number of simulations run for each  $w_D$ .

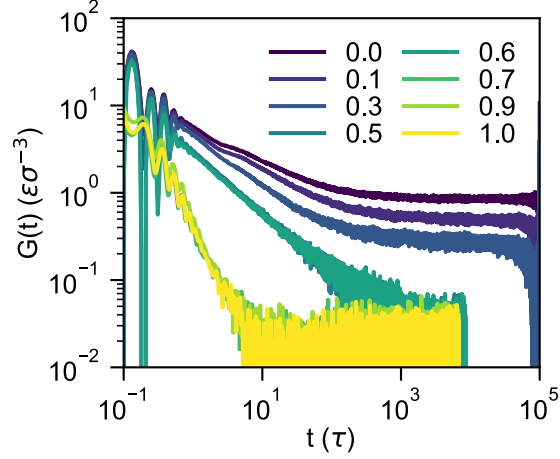


Figure S4: Shear modulus  $G(t)$ . It can be seen that materials optimized with a mechanical emphasis have plateau  $G(t)$  at intermediate times, suggesting that these materials are at least viscoelastic.

In Figure S4, it can be seen that polymer electrolytes with high emphasis on their mechanical properties (low  $w_D$ ) seem to be viscoelastic (plateau shear moduli  $G(t)$  at long times). In combination with results shown in the main text of this work, these results suggest that the optimization of the proxy performance  $f_p(\mathbf{x})$  indeed yielded a trade off between the true performance metrics with a potential increase in shear elasticity at low  $w_D$ .

## References

- (1) Rasmussen, C. E.; Williams, C. K. I. *Gaussian Processes for Machine Learning*; The MIT Press: Cambridge, Massachusetts, 2006.
- (2) González, J.; Dai, Z.; Hennig, P.; Lawrence, N. Batch bayesian optimization via local penalization. *Proc. 19th Int. Conf. Artif. Intell. Stat. AISTATS 2016* **2016**, 648–657.