Supplementary Information

Nucleic Acid Sequence Design via Efficient Ensemble Defect Optimization

Joseph N. Zadeh¹, Brian R. Wolfe¹, Niles A. Pierce^{1,2}

¹Department of Bioengineering, California Institute of Technology, Pasadena, CA 91125 ²Department of Applied & Computational Mathematics, California Institute of Technology, Pasadena, CA 91125 Correspondence to: niles@caltech.edu

```
DESIGNSEQ(s)
      m_{\mathrm{leafopt}} \leftarrow 0
      \phi, n \leftarrow \mathsf{OPTIMIZELEAF}(s)
      while n > f_{\rm stop} |s| and m_{\rm leafopt} < M_{\rm leafopt}
              \hat{\phi}, \hat{n} \leftarrow \text{OptimizeLeaf}(s)
             \text{if } \hat{n} < n
                    \phi, n \leftarrow \hat{\phi}, \hat{n}
             m_{\text{leafopt}} \leftarrow m_{\text{leafopt}} + 1
      return \phi
OPTIMIZELEAF(s)
      m_{\text{unfavorable}} \leftarrow 0
      \gamma_{\text{unfavorable}} \leftarrow \emptyset
      \phi \leftarrow \text{INITSEQ}(s)
      n \leftarrow \texttt{EnsembleDefect}(\phi, s)
      while n > f_{\rm stop} |s| and m_{\rm unfavorable} < M_{\rm unfavorable} |s|
              \xi, \hat{\phi} \leftarrow \text{UniformMutationSampling}(\phi, s)
             if \xi \in \gamma_{\text{unfavorable}}
                     m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                     \hat{n} \leftarrow \text{EnsembleDefect}(\hat{\phi}, s)
                     if \hat{n} < n
                             \phi, n \leftarrow \hat{\phi}, \hat{n}
                             m_{\text{unfavorable}} \leftarrow 0
                             \gamma_{\text{unfavorable}} \leftarrow \emptyset
                             m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                             \gamma_{\text{unfavorable}} \leftarrow \gamma_{\text{unfavorable}} \bigcup \xi
      return \phi, n
```

```
Algorithm S1: Single-scale ensemble defect optimization with uniform mutation sampling.
```

```
DESIGNSEQ(s)
      m_{\text{leafopt}} \leftarrow 0
       \phi, n \leftarrow \mathsf{OPTIMIZELEAF}(s)
       while n > f_{\rm stop} |s| and m_{\rm leafopt} < M_{\rm leafopt}
               \hat{\phi}, \hat{n} \leftarrow \text{OPTIMIZELEAF}(\hat{\phi}, s)
              \text{if } \hat{n} < n
                      \phi, n \leftarrow \hat{\phi}, \hat{n}
              m_{\text{leafopt}} \leftarrow m_{\text{leafopt}} + 1
       return \phi
OPTIMIZELEAF(s)
      m_{\text{unfavorable}} \leftarrow 0
       \gamma_{\text{unfavorable}} \leftarrow \emptyset
       \phi \leftarrow \text{INITSEQ}(s)
      n \leftarrow \text{EnsembleDefect}(\phi, s)
       while n > f_{\rm stop}|s| and m_{\rm unfavorable} < M_{\rm unfavorable}|s|
              \xi, \hat{\phi} \leftarrow \text{WeightedMutationSampling}(\phi, s, n_1, \dots, n_{|s|})
               if \xi \in \gamma_{\mathrm{unfavorable}}
                      m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                      \hat{n} \leftarrow \text{EnsembleDefect}(\hat{\phi}, s)
                      \text{if } \hat{n} < n
                             \phi, n \leftarrow \hat{\phi}, \hat{n}
                             m_{\text{unfavorable}} \leftarrow 0
                              \gamma_{\text{unfavorable}} \leftarrow \emptyset
                              m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                              \gamma_{\text{unfavorable}} \leftarrow \gamma_{\text{unfavorable}} \cup \xi
       return \phi, n
```

Algorithm S2: Single-scale ensemble defect optimization with defect-weighted mutation sampling.

```
DESIGNSEQ(\phi, s, n, k)
                                                                                                                                                UPDATECHILDREN(k, a, b)
                                                                                                                                                      if HasChildren(k)
      a \leftarrow \text{DEPTH}(k)
                                                                                                                                                              n^{k_l,a} \leftarrow n^{k_l,b}
n^{k_r,a} \leftarrow n^{k_r,b}
      if HASCHILDREN(k)
              m_{\text{reopt}} \leftarrow 0
             if n = \emptyset
                                                                                                                                                              UPDATECHILDREN(k_l, a, b)
                     \phi_l \leftarrow \text{DESIGNSEQ}(\emptyset, s_{l+}, \emptyset, k_l)
                                                                                                                                                               UPDATECHILDREN(k_r, a, b)
                      \phi_r \leftarrow \text{DESIGNSEQ}(\emptyset, s_{r+}, \emptyset, k_r)
             else
                                                                                                                                                OPTIMIZELEAF(s)
                     UPDATECHILDREN(k, a, a - 1)
                                                                                                                                                       m_{\rm unfavorable} \leftarrow 0
                     child, \phi \leftarrow \text{UniformChildSampling}(\phi, s, n_l, n_r)
                                                                                                                                                       \gamma_{\text{unfavorable}} \leftarrow \emptyset
                     \phi_{\text{child}} \leftarrow \text{DESIGNSEQ}(\phi_{\text{child}+}, s_{\text{child}+}, n_{\text{child}+}, k_{\text{child}})
                                                                                                                                                       \phi \leftarrow \text{INITSEQ}(s)
              n^{k,a} \leftarrow \texttt{EnsembleDefect}(\phi,s)
             n^{n,c} \leftarrow \text{ENSEMBLEDGE STATES}, \\ \text{UPDATECHILDREN}(k, a, a + 1) \\ \text{while } n^{k,a} > \max(f_{\text{stop}}|s_l|, n^{k_l,a}_{\text{native}}) + \max(f_{\text{stop}}|s_r|, n^{k_r,a}_{\text{native}})
                                                                                                                                                       n \leftarrow \text{EnsembleDefect}(\phi, s)
                                                                                                                                                       while n > f_{\rm stop}|s| and m_{\rm unfavorable} < M_{\rm unfavorable}|s|
                                                                                                                                                              \xi, \hat{\phi} \leftarrow \text{UniformMutationSampling}(\phi, s)
                             and m_{\text{reopt}} < M_{\text{reopt}}
                                                                                                                                                               if \xi \in \gamma_{\text{unfavorable}}
                     \textbf{child}, \hat{\phi} \leftarrow \textbf{UniformChildSampling}(\phi, s, n_l^{k,a}, n_r^{k,a})
                                                                                                                                                                      m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                     \hat{\phi}_{\text{child}} \leftarrow \text{DESIGNSEQ}(\phi_{\text{child+}}, s_{\text{child+}}, n_{\text{child+}}^{k, a}, k_{\text{child}})
                     \hat{n} \leftarrow \text{EnsembleDefect}(\hat{\phi}, s)
                                                                                                                                                                      \hat{n} \leftarrow \text{EnsembleDefect}(\hat{\phi}, s)
                     \text{if } \hat{n} < n^{k,a}
                                                                                                                                                                      if \hat{n} < n
                             \phi, n^{k,a} \leftarrow \hat{\phi}, \hat{n}
                                                                                                                                                                              \phi, n \leftarrow \hat{\phi}, \hat{n}
                             UPDATECHILDREN(k, a, a + 1)
                                                                                                                                                                              m_{\text{unfavorable}} \leftarrow 0
                     m_{\text{reopt}} \leftarrow m_{\text{reopt}} + 1
                                                                                                                                                                              \gamma_{\text{unfavorable}} \leftarrow \emptyset
      else
                                                                                                                                                                      else
             m_{\text{leafopt}} \leftarrow 0
                                                                                                                                                                              m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
              \phi, n^{k,a} \leftarrow \text{OPTIMIZELEAF}(s)
                                                                                                                                                                              \gamma_{\text{unfavorable}} \leftarrow \gamma_{\text{unfavorable}} \cup \xi
              while n^{k,a} > f_{\text{stop}}|s| and m_{\text{leafopt}} < M_{\text{leafopt}}
                                                                                                                                                      return \phi, n
                     \hat{\phi}, \hat{n} \leftarrow \text{OPTIMIZELEAF}(s)
                     if \hat{n} < n^{k,a}
                             \phi, n^{k,a} \leftarrow \hat{\phi}, \hat{n}
                     m_{\text{leafopt}} \leftarrow m_{\text{leafopt}} + 1
      return \phi_{\mathrm{native}}
```

Algorithm S3: Hierarchical ensemble defect optimization with uniform sampling. Pseudocode conventions follow those of Algorithm 1.

```
DESIGNSEQ(s)
       \begin{aligned} m_{\text{leafopt}} &\leftarrow 0 \\ \phi, \pi &\leftarrow \text{OptimizeLeaf}(s) \end{aligned}
       while \pi > f_{\rm stop} and m_{\rm leafopt} < M_{\rm leafopt}
                \hat{\phi}, \hat{\pi} \leftarrow \text{OPTIMIZELEAF}(s)
                if \hat{\pi} < \pi
                         \phi, \pi \leftarrow \hat{\phi}, \hat{\pi}
                m_{\text{leafopt}} \leftarrow m_{\text{leafopt}} + 1
       return \phi
OPTIMIZELEAF(s)
       m_{\text{unfavorable}} \leftarrow 0\gamma_{\text{unfavorable}} \leftarrow \emptyset
       \phi \leftarrow \text{INITSEQ}(s)
       \pi \leftarrow PROBABILITYDEFECT(\phi, s)
       while \pi > f_{\rm stop} and m_{\rm unfavorable} < M_{\rm unfavorable} |s|
                \xi, \hat{\phi} \leftarrow \text{UniformMutationSampling}(\phi, s)
                \text{if } \xi \in \gamma_{\text{unfavorable}}
                         m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                 else
                          \hat{\pi} \leftarrow \text{ProbabilityDefect}(\hat{\phi}, s)
                         if \hat{\pi} < \pi
                                  \phi, \pi \leftarrow \hat{\phi}, \hat{\pi}
                                  m_{\rm unfavorable} \leftarrow 0
                                  \gamma_{\text{unfavorable}} \leftarrow \emptyset
                         else
                                 \begin{aligned} & m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1 \\ & \gamma_{\text{unfavorable}} \leftarrow \gamma_{\text{unfavorable}} \bigcup \xi \end{aligned}
       return \phi, \pi
```

Algorithm S4: Single-scale probability defect optimization with uniform mutation sampling.

```
DesignSeq(\phi, s, \mu, k)
                                                                                                                                                      UPDATECHILDREN(k, a, b)
      a \leftarrow \text{DEPTH}(k)
                                                                                                                                                             if HasChildren(k)
                                                                                                                                                                     \mu^{k_l,a} \leftarrow \mu^{k_l,b}
\mu^{k_r,a} \leftarrow \mu^{k_r,b}
       if HasChildren(k)
              m_{\text{reopt}} \leftarrow 0
              if \mu = \emptyset
                                                                                                                                                                     UPDATECHILDREN(k_l, a, b)
                      \phi_l \leftarrow \text{DESIGNSEQ}(\emptyset, s_{l+}, \emptyset, k_l)
                                                                                                                                                                      UPDATECHILDREN(k_r, a, b)
                       \phi_r \leftarrow \text{DESIGNSEQ}(\emptyset, s_{r+}, \emptyset, k_r)
                       UPDATECHILDREN(k, a, a - 1)
                      child, \phi \leftarrow \text{WeightedChildSampling}(\phi, s, \mu_l, \mu_r)
                                                                                                                                                      OPTIMIZELEAF(s)
                      \phi_{\text{child}} \leftarrow \text{DESIGNSEQ}(\phi_{\text{child}+}, s_{\text{child}+}, \mu_{\text{child}+}, k_{\text{child}})
                                                                                                                                                             m_{\text{try}} \leftarrow 0
              \mu^{k,a} \leftarrow \mathsf{MFEDEFECT}(\phi,s)
                                                                                                                                                             m_{\text{unfavorable}} \leftarrow 0
              {\tt UPDATECHILDREN}(k,a,a+1)
                                                                                                                                                             \gamma_{\text{unfavorable}} \leftarrow \emptyset
              while \mu^{k,a} > \max(f_{\text{stop}}|s_l|, \mu_{\text{native}}^{k_l,a}) + \max(f_{\text{stop}}|s_r|, \mu_{\text{native}}^{k_r,a})
                                                                                                                                                             \phi \leftarrow \text{INITSEQ}(s)
                              and m_{\text{reopt}} < M_{\text{reopt}}
                                                                                                                                                             \mu \leftarrow \text{MFEDEFECT}(\phi, s)
                      \hat{\mu}_i \leftarrow \mu_i^{k,a} + \epsilon \quad \forall i \in \{1, \dots, |s|\}
                                                                                                                                                             while \mu > f_{\rm stop}|s| and m_{\rm unfavorable} < M_{\rm unfavorable}|s|
                      child, \hat{\phi} \leftarrow \text{WEIGHTEDCHILDSAMPLING}(\phi, s, \hat{\mu}_l, \hat{\mu}_r)
                                                                                                                                                                     and m_{\text{try}} < M_{\text{try}}

\hat{\mu}_i \leftarrow \mu_i + \epsilon \quad \forall i \in \{1, \dots, |s|\}
                      \hat{\phi}_{\text{child}} \leftarrow \text{DESIGNSEQ}(\phi_{\text{child}+}, s_{\text{child}+}, \hat{\mu}_{\text{child}+}, k_{\text{child}})
                      \hat{\mu} \leftarrow \text{MFEDEFECT}(\hat{\phi}, s)
                                                                                                                                                                     \xi, \hat{\phi} \leftarrow \text{WeightedMutationSampling}(\phi, s, \hat{\mu}_1, \dots, \hat{\mu}_{|s|})
                      \text{if } \hat{\mu} < \mu^{k,a}
                                                                                                                                                                     if \xi \in \gamma_{\mathrm{unfavorable}}
                              \phi, \overset{\cdot}{\mu}{}^{k,a} \leftarrow \hat{\phi}, \hat{\mu}
                                                                                                                                                                             m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                              UPDATECHILDREN(k, a, a + 1)
                      m_{\text{reopt}} \leftarrow m_{\text{reopt}} + 1
                                                                                                                                                                              \hat{\mu} \leftarrow \mathsf{MFEDEFECT}(\hat{\phi}, s)
       else
                                                                                                                                                                              if \hat{\mu} < \mu or ACCEPTUNFAVORABLE(f_{\text{accept}})
              m_{\text{leafopt}} \leftarrow 0
                                                                                                                                                                                     \phi, \mu \leftarrow \hat{\phi}, \hat{\mu}
              \phi, \mu^{k, \hat{a}} \leftarrow \text{OptimizeLeaf}(s)
                                                                                                                                                                                     m_{\text{unfavorable}} \leftarrow 0
              while \mu^{k,a} > f_{\mathrm{stop}} |s| and m_{\mathrm{leafopt}} < M_{\mathrm{leafopt}}
                                                                                                                                                                                     \gamma_{\text{unfavorable}} \leftarrow \emptyset
                      \hat{\phi}, \hat{\mu} \leftarrow \text{OPTIMIZELEAF}(s)
                      if \hat{\mu} < \mu^{k,a}
                                                                                                                                                                                      m_{\text{unfavorable}} \leftarrow m_{\text{unfavorable}} + 1
                              \phi, \mu^{k,a} \leftarrow \hat{\phi}, \hat{\mu}
                                                                                                                                                                                     \gamma_{\text{unfavorable}} \leftarrow \gamma_{\text{unfavorable}} \cup \xi
                      m_{\text{leafopt}} \leftarrow m_{\text{leafopt}} + 1
                                                                                                                                                                     m_{\text{try}} \leftarrow m_{\text{try}} + 1
      return \phi_{\mathrm{native}}
                                                                                                                                                             return \phi, \mu
```

Algorithm S5: Hierarchical MFE defect optimization with defect-weighted sampling. During leaf optimization, we employ defect-weighted mutation sampling, selecting nucleotide i as a mutation candidate with probability $(\mu_i^{k,a} + \epsilon)/(\mu^{k,a} + \epsilon|s|)$. Adding ϵ to each defect contribution ensures that all bases (even those with $\mu_i^{k,a} = 0$) are subject to mutation with a non-zero probability. During leaf optimization, fraction f_{accept} of unfavorable candidate mutations are accepted to assist in escaping from local minima. The leaf stop condition is $\mu^{k,a} < f_{\text{stop}}|s|$; the parental stop condition is $\mu^{k,a} < \max(f_{\text{stop}}|s_l|,\mu_{\text{native}}^{k_l,a}) + \max(f_{\text{stop}}|s_r|,\mu_{\text{native}}^{k_r,a})$. Because some unfavorable mutations are accepted, the total number of mutation attempts during a leaf optimization is limited to M_{try} . Calculations are performed with defaults values: $\epsilon = 0.1$, $f_{\text{accept}} = 0.2$, $M_{\text{try}} = 5000$. Pseudocode conventions follow those of Algorithm 1.

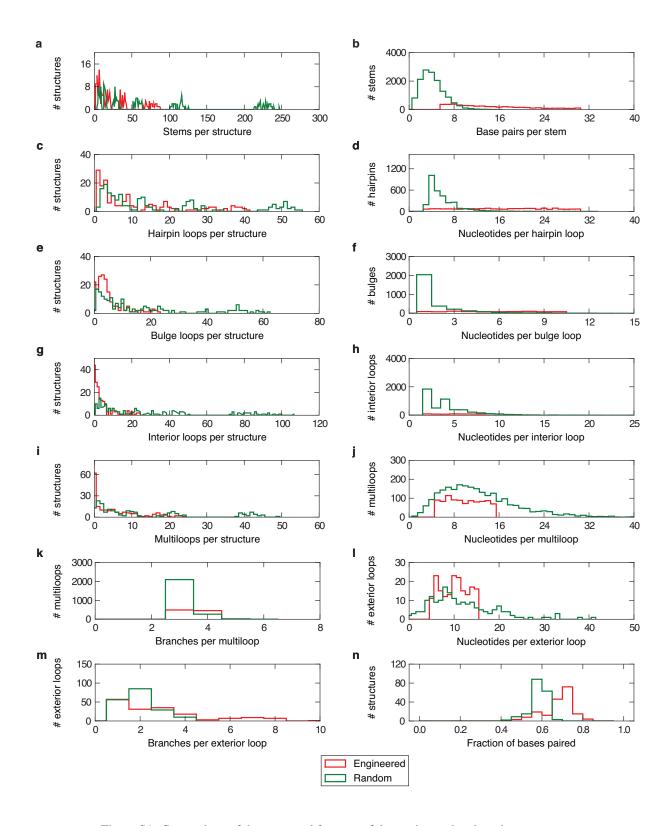


Figure S1: Comparison of the structural features of the engineered and random test sets.

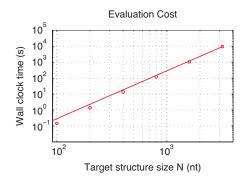


Figure S2: Computational cost, $c_{\rm eval}(N) = \Theta(N^3)$, of a single evaluation of the ensemble defect, $n(\phi,s)$, for the full sequence and target structure. Each data point represents the median over all sequences for a particular value of N. The line depicts a slope of three, suggesting empirically that the dynamic program is operating approximately within the asymptotic regime for this range of N. RNA design at $37^{\circ}{\rm C}$ on the engineered test set.

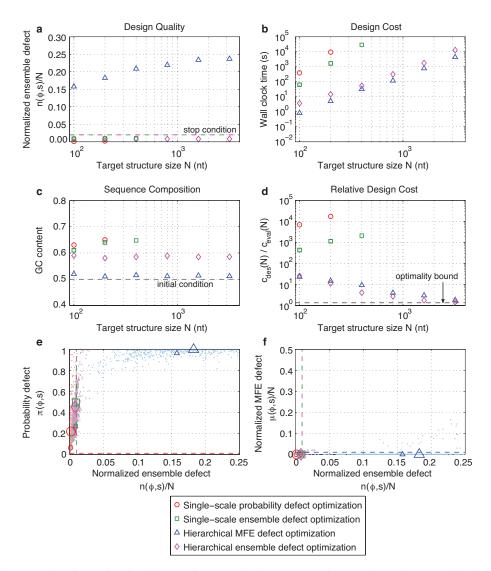


Figure S3: Comparison to algorithms inspired by previous publications. a) Design quality. The ensemble defect stop condition is depicted as a dashed line. b) Design cost. c) Sequence composition. The initial GC content is depicted as a dashed line. d) Cost of sequence design relative to a single evaluation of the objective function. The optimality bound is depicted as a dashed line. e,f) Evaluation of each sequence design using three objective functions. Dots represent independent designs. Symbols denote medians for each value of $N \in \{100, 200\}$ (symbol size increases with N). RNA design at 37°C on the random test set.