Article

Algorithm for optimized mRNA design improves stability and immunogenicity

https://doi.org/10.1038/s41586-023-06127-z

Received: 12 March 2022

Accepted: 25 April 2023

Published online: 2 May 2023

Open access

Check for updates

He Zhang[1,2,11], Liang Zhang[1,2,8,11], Ang Lin[3,8,11], Congcong Xu[3,11], Ziyu Li[1], Kaibo Liu[1,2], Boxiang Liu[1,9], Xiaopin Ma[3], Fanfan Zhao[3], Huiling Jiang[3], Chunxiu Chen[3], Haifa Shen[3], Hangwen Li[3]✉, David H. Mathews[4,5,6,7]✉, Yujian Zhang[3,10]✉ & Liang Huang[1,2,7,11]✉

Messenger RNA (mRNA) vaccines are being used to combat the spread of COVID-19 (refs. 1–3), but they still exhibit critical limitations caused by mRNA instability and degradation, which are major obstacles for the storage, distribution and efficacy of the vaccine products[4]. Increasing secondary structure lengthens mRNA half-life, which, together with optimal codons, improves protein expression[5]. Therefore, a principled mRNA design algorithm must optimize both structural stability and codon usage. However, owing to synonymous codons, the mRNA design space is prohibitively large—for example, there are around 2.4×10^{632} candidate mRNA sequences for the SARS-CoV-2 spike protein. This poses insurmountable computational challenges. Here we provide a simple and unexpected solution using the classical concept of lattice parsing in computational linguistics, where finding the optimal mRNA sequence is analogous to identifying the most likely sentence among similar-sounding alternatives[6]. Our algorithm LinearDesign finds an optimal mRNA design for the spike protein in just 11 minutes, and can concurrently optimize stability and codon usage. LinearDesign substantially improves mRNA half-life and protein expression, and profoundly increases antibody titre by up to 128 times in mice compared to the codon-optimization benchmark on mRNA vaccines for COVID-19 and varicella-zoster virus. This result reveals the great potential of principled mRNA design and enables the exploration of previously unreachable but highly stable and efficient designs. Our work is a timely tool for vaccines and other mRNA-based medicines encoding therapeutic proteins such as monoclonal antibodies and anti-cancer drugs[7,8].

mRNA vaccines[9,10] have been recognized as viable tools to limit the spread of COVID-19 owing to their scalable production, safety and efficacy[1–3]. However, mRNA molecules are chemically unstable and prone to degrade, which leads to insufficient protein expression[5], and, in turn, compromised immunogenicity and druggability. This instability has also become a major obstacle in the storage and distribution of the vaccine, requiring the use of cold-chain technologies that hinders its use in developing countries[4]. Thus an mRNA molecule with enhanced stability is desirable, which would potentially have greater potency and favourable clinical efficacy.

Although it remains difficult to model chemical stability, previous work has established its correlation with RNA secondary structure, as quantified by the well-studied thermodynamic folding stability. Improving this structural stability, combined with optimal codon usage, leads to increased protein expression[5]. Therefore, a principled mRNA design algorithm must optimize two factors—structural stability and codon usage—to enhance protein expression.

However, the mRNA design problem (we consider only the coding region in this work) is extremely challenging owing to the exponentially large search space. Each amino acid is encoded by a triplet codon—that is, three adjacent nucleotides—but owing to redundancies in the genetic code, most amino acids have multiple codons; there are 4^3 (that is, 64) codons for the 20 common naturally occurring amino acids. This results in a prohibitively large number of candidates for any protein sequence. For example, the spike protein of SARS-CoV-2 has 1,273 amino acids and can therefore be encoded by approximately 2.4×10^{632} mRNA sequences (Fig. 1a). This poses an insurmountable computational challenge and rules out enumeration, which would take 10^{616} billion years for the spike protein (Fig. 1b). Conversely, codon optimization[11,12], the conventional approach to mRNA design, optimizes codon usage but barely improves stability, leaving out the huge space of highly stable mRNAs. Optimizing GC content has a similar effect as it correlates with codon usage in vertebrates[13]. As a result, the vast majority of highly stable designs remains unexplored.

[1]Baidu Research USA, Sunnyvale, CA, USA. [2]School of EECS, Oregon State University, Corvallis, OR, USA. [3]StemiRNA Therapeutics, Shanghai, China. [4]Department of Biochemistry and Biophysics, University of Rochester Medical Center, Rochester, NY, USA. [5]Center for RNA Biology, University of Rochester Medical Center, Rochester, NY, USA. [6]Department of Biostatistics and Computational Biology, University of Rochester Medical Center, Rochester, NY, USA. [7]Coderna.ai, Inc., Sunnyvale, CA, USA. [8]Present address: Vaccine Center, School of Basic Medicine and Clinical Pharmacy, China Pharmaceutical University, Nanjing, China. [9]Present address: Department of Pharmacy, National University of Singapore, Singapore, Singapore. [10]Present address: Gaithersburg, MD, USA. [11]These authors contributed equally: He Zhang, Liang Zhang, Ang Lin, Congcong Xu, Liang Huang. ✉e-mail: lihangwen@stemirna.com; david_mathews@urmc.rochester.edu; yujianzhang@yahoo.com; liang.huang.sh@gmail.com

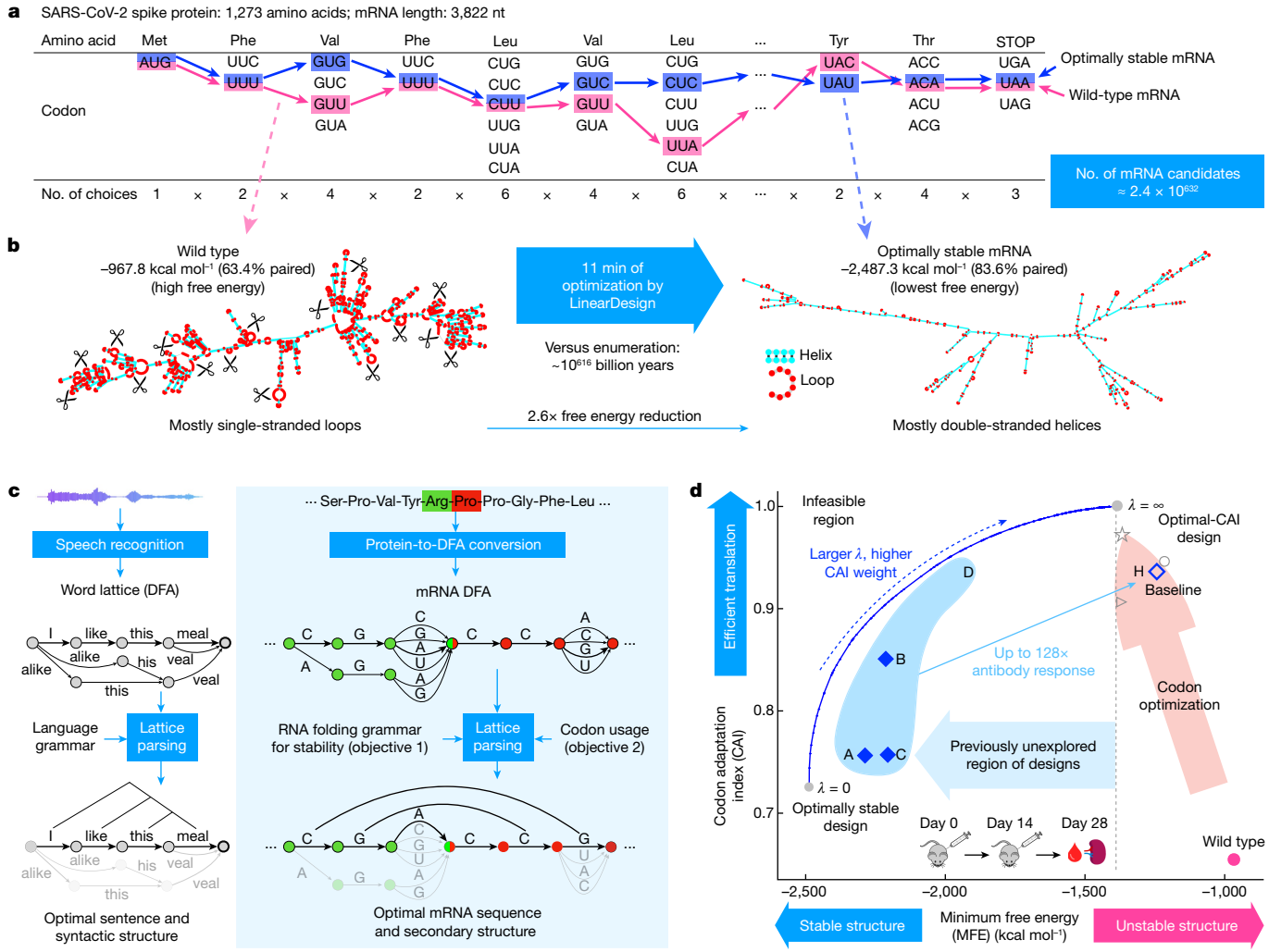


Fig. 1 | Overview of mRNA coding region design for both stability and codon optimality using SARS-CoV-2 spike protein as an example. a, Due to codon degeneracy and combinatorial explosion, there are around 2.4×10^{632} possible mRNA sequences encoding the spike protein. Enumerating every possible sequence would take around 10^{616} billion years. The pink and blue paths represent the wild-type and the optimally stable (lowest free energy) sequences, respectively. nt, nucleotides. **b**, The secondary structures of wild-type (left) and optimally stable (right) spike mRNAs. The wild-type mRNA is mostly single-stranded and thus prone to degradation in loop regions (red), whereas the optimally stable mRNA is mostly double-stranded. Optimization using LinearDesign takes around 11 min. **c**, The application of DFA and lattice parsing in computational linguistics (left) and its adaptation to mRNA design (right). An mRNA DFA (analogous to a word lattice) compactly encodes all mRNA candidates, which are folded simultaneously by lattice parsing to find the optimal mRNA (Fig. 2). **d**, Two-dimensional visualization of the mRNA design space, with stability (represented by MFE) on the *x* axis and codon optimality (represented by CAI) on the *y* axis. The standard mRNA design method of codon optimization improves codon usage (pink arrow) but is unable to explore the high-stability region (left of the dashed line); this standard approach is exemplified by the COVID-19 mRNA vaccine products BNT-162b2 (BioNTech-Pfizer, circle), mRNA-1273 (Moderna, star) and CVnCoV/CV2CoV (CureVac, wedge). LinearDesign jointly optimizes stability and codon optimality (blue curve, with λ being the weight assigned to codon optimality). We selected seven mRNA designs (four (A–D) are shown here) and a codon-optimized baseline (H) for in vitro and in vivo experiments (Fig. 4).

Here we describe LinearDesign, an algorithm that addresses this challenge by adapting the classical concept of lattice parsing[6] in computational linguistics (Fig. 1c). We show that finding the optimal mRNA among the vast space of candidates is analogous to finding the most likely sentence among many similar-sounding alternatives. More specifically, we formulate the mRNA design space using a deterministic finite-state automaton (DFA), similar to a word lattice[6], which compactly encodes exponentially many mRNA candidates. We then use lattice parsing to find the most stable mRNA in the DFA, or the optimal balance between stability and codon optimality in a weighted DFA. This unexpected connection to natural language enables an efficient algorithm that scales quadratically with the mRNA sequence length in practice. In this sense, our work transforms the enormous design space into an advantage—providing freedom of design—rather than an obstacle.

Compared to the codon-optimized benchmark, our COVID-19 and varicella-zoster virus (VZV) mRNA vaccines substantially improve chemical stability in vitro, protein expression in cells and immunogenicity in vivo. In particular, our COVID-19 vaccines achieved up to 128 times the antibody response of the codon-optimized benchmark in mice. This result reveals the great potential of principled mRNA design, and enables the exploration of these previously unreachable but highly stable and efficient designs. Our work provides a timely and promising tool for the design of mRNA vaccines and other mRNA-based medicines[14] encoding therapeutic proteins including monoclonal antibodies[7] and anti-cancer drugs[8].

Formulations and algorithms

Previous work[5] established two main objectives for mRNA design, stability and codon optimality, which synergize to increase protein expression. To optimize for stability, given a protein sequence, we aim to find the mRNA sequence that has the lowest minimum-free-energy change (MFE) among all possible mRNA sequences encoding that protein; that is, for each candidate mRNA sequence, we find its MFE structure among all its possible secondary structures using the standard RNA folding energy model[15,16] and then choose the sequence whose MFE energy is the lowest. This is thus a minimization within a minimization (Extended Data Fig. 1a). This method would take billions of years, thus an efficient algorithm without enumeration is needed.

We also aim to jointly optimize mRNA stability and codon optimality. Codon optimality is often measured by the codon adaptation index[17] (CAI), which is defined as the geometric mean of the relative adaptiveness of each codon in the mRNA. Because CAI is between 0 and 1 but MFE is generally proportional to the sequence length, we multiply the logarithm of CAI by the number of codons in the mRNA and use the hyper-parameter CAI weight (λ) to balance MFE and CAI ($\lambda = 0$ being MFE-only). The combined objective is MFE $- \lambda|\mathbf{p}| \log$ CAI, where $|\mathbf{p}|$ is the protein length. See Methods, 'Optimization objectives' and Extended Data Fig. 1b for details.

We next describe our solution to these two optimization problems with two ideas borrowed from natural language: DFA (lattice) representation and lattice parsing.

Lattice representation for mRNA design space

Inspired by the word lattice representation of ambiguities in computational linguistics (Extended Data Fig. 2a), we represent the choice of codons for each amino acid using a similar lattice—more formally, a DFA, which is a directed graph with nucleotide-labelled edges (Fig. 2a and Extended Data Fig. 1c; see Methods, 'DFA representations for codons and mRNA candidate sequences' for formal definitions). After building a codon DFA for each amino acid in the protein sequence, we concatenate them into a single mRNA DFA, where each path between the start and final states represents a possible mRNA sequence encoding that protein (Fig. 2b and Extended Data Fig. 1d).

Lattice parsing

RNA folding is known to be equivalent to natural language parsing, where a stochastic context-free grammar (SCFG) can represent the folding energy model[18] (Extended Data Fig. 1e,f). For mRNA design, the hard question is how all the mRNA sequences in the DFA can be folded together. We borrow the idea of lattice parsing[6,19], which generalizes single-sequence parsing to handle all sentences in the lattice simultaneously to find the most likely one (Fig. 1c and Extended Data Fig. 2). Similarly, we use lattice parsing to fold all sequences in the mRNA DFA simultaneously to find the most stable one (Fig. 2b and Extended Data Fig. 1g,h). Note that lattice parsing is also an instance of dynamic programming, but over a much larger search space, and single-sequence folding can be viewed as a special case of lattice parsing with a single-chain DFA. This process can also be interpreted as the SCFG–DFA intersection (Extended Data Fig. 1a) where the SCFG scores for stability and the DFA demarcates the set of candidates. The runtime of this algorithm scales cubically with the mRNA sequence length (Methods, 'SCFG, lattice parsing and intersection'), but for practical applications it scales quadratically (Fig. 3a).

Lattice parsing with weighted DFAs

We now extend DFAs to weighted DFAs to integrate codon optimality on edge weights. Since our joint optimization formulation factors CAI onto the relative adaptiveness $w(c)$ of each individual codon c, we set edge weights in each codon DFA so that a codon c has path cost $-\log w(c)$, which can be interpreted as the "amount of deviation" from the optimal codon. Then in a weighted mRNA DFA, the cost of each start-end path is the sum of $-\log w(c)$ for each codon c in the corresponding mRNA, which is proportional to its −log CAI (Fig. 2d). Now lattice parsing takes a stochastic grammar (for stability) and a weighted DFA (for codon usage) and solves the joint optimization with optimality guarantee, which can be viewed as the weighted intersection[20] between an SCFG and a weighted DFA (Extended Data Fig. 1b and Methods, 'Weighted DFA for CAI integration').

Expressiveness of DFAs

Our DFA framework is sufficiently general that it can also represent alternative genetic codes, modified nucleotides and coding constraints. For details, see Methods, 'DFAs for other genetic codes, coding constraints and modified nucleotides', Extended Data Fig. 3 and Supplementary Fig. 5.

Linear-time approximation

The exact design algorithm might still be slow for long sequences. Additionally, suboptimal designs may also be worth exploring for wet laboratory experiments, owing to the many other factors involved in mRNA design besides stability and codon usage. We therefore developed an approximate search version that runs in linear time using beam search, keeping only the top b most promising items per step (where b is the beam size), inspired by our previous work LinearFold[21].

Related work

Two previous studies also tackled the problem of 'most stable mRNA design' (our objective 1) via dynamic programming, but using specialized extensions of the Zuker algorithm[22,23] that cannot incorporate codon optimality (objective 2). By contrast, we established the connection between mRNA design and lattice parsing from computational linguistics. This connection enabled a simpler and more generalizable algorithm that can jointly optimize codon usage with a novel objective function that factors CAI onto individual codons. We also verified these algorithmic designs in vivo, showing substantial improvements for two mRNA vaccines (Figs. 4 and 5). See Methods, 'The LinearDesign algorithm' and 'Related work' for details.

In silico results and analysis

Figure 3a benchmarks the runtime of LinearDesign on UniProt proteins[24]. LinearDesign was shown in a combination of two optimization objectives: MFE-only (objective 1) versus MFE and CAI (objectives 1 and 2), and two search modes: exact search versus beam search ($b = 500$). Empirically, LinearDesign scales quadratically with mRNA sequence length n for practical applications ($n < 10{,}000$ nt) thanks to the DFA representation and lattice parsing (Supplementary Figs. 7 and 8). Next, our CAI-integrated exact search ($\lambda = 4$) had the same empirical complexity, and was only around 15% slower than the MFE-only version thanks to the convenience of our DFA representation for adding CAI. Last, our beam search version ($b = 500$) further speeds up the design and scales linearly with sequence length, taking only 2.7 min (versus 10.7 min for exact search) on the SARS-CoV-2 spike protein (for MFE-only), with an approximation error (percentage energy gap, defined as $(1 - \mathrm{MFE}_{\mathrm{approx\_design}}/\mathrm{MFE}_{\mathrm{exact\_design}}) \times 100\%$) of 1.2%, where the subscripts indicate approximate or exact design. In fact, as sequences get longer, this percentage stabilizes, suggesting that beam search quality does not degrade with sequence length (Supplementary Fig. 9).

For a GC-favouring codon preference (such as in humans), the conventional codon-optimization method does improve stability, but only slightly, since the codon-optimization direction (pink arrows) are largely orthogonal to the stability optimization direction (blue arrows) (Fig. 3b,c). By contrast, our LinearDesign can directly optimize stability and find the optimally stable mRNAs. On both the SARS-CoV-2 spike protein and the VZV gE protein, the lowest MFEs ($\lambda = 0$) are 1.8 times

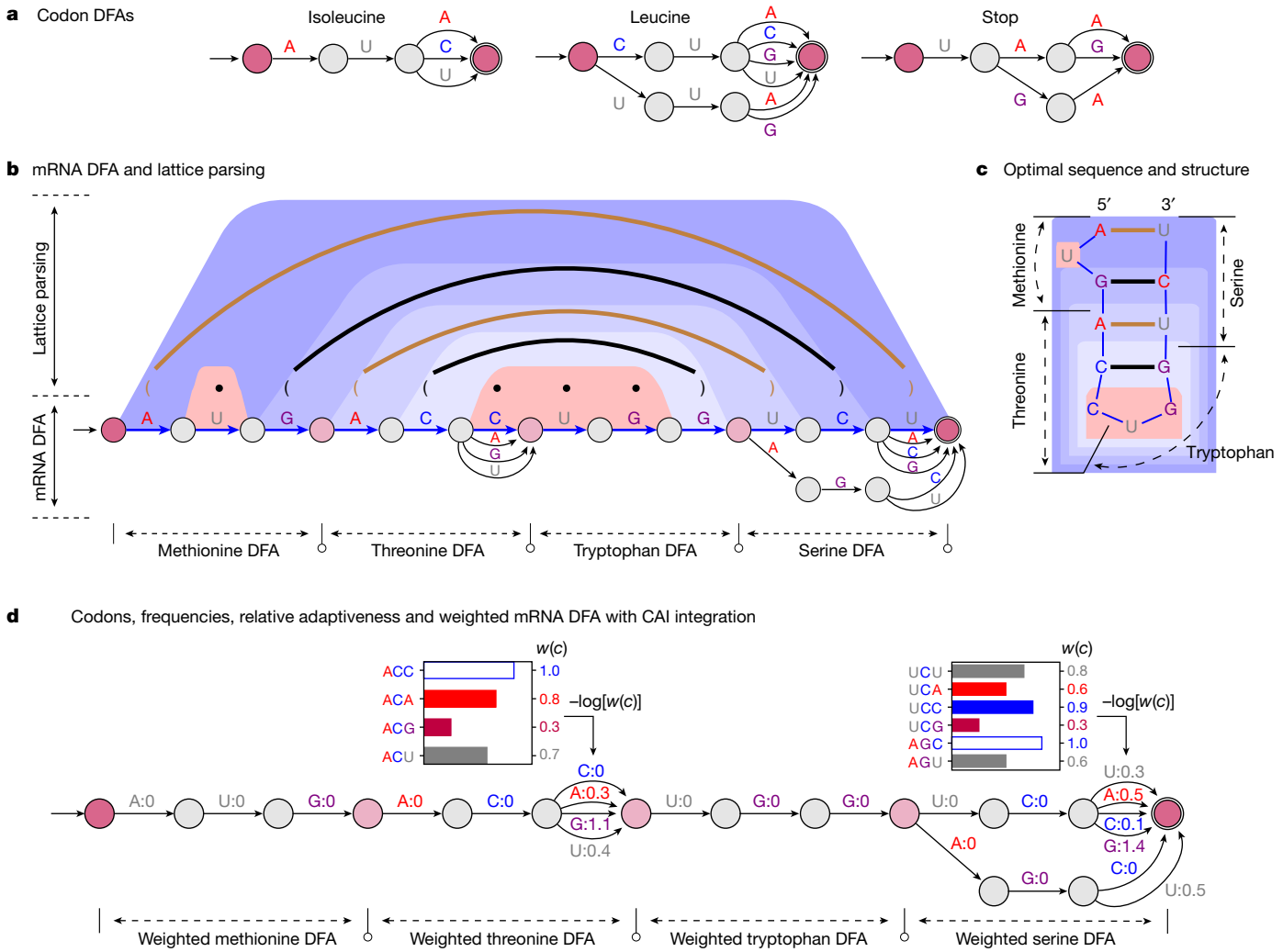


Fig. 2 | Illustration of the LinearDesign algorithm. a, Codon DFAs. **b**, An mRNA DFA (bottom) and lattice parsing on that DFA (top). In the DFA, the optimal mRNA sequence under a simplified energy model is shown as the blue path, together with its optimal structure shown in the dot-bracket format (dots indicate unpaired, and brackets indicate base pairs). In lattice parsing, the brown and black arcs also depict base pairs (two GC pairs and two AU pairs), and the trapezoidal shaded areas depict the decomposition of the optimal structure. Among all mRNA sequences encoded in the DFA, lattice parsing finds the optimal sequence with its optimal structure, achieving the lowest free energy under this energy model, where GC and AU pairs have energies of −3 and −2 kcal mol^{-1}, respectively (Extended Data Fig. 1e). Note that here we use the simplified energy model for illustration, but our implementation uses the nearest-neighbour energy model. **c**, Another illustration of the optimal sequence and secondary structure in **b**. **d**, Joint optimization between stability and codon optimality by integrating the codon optimality in weighted DFAs. Top, the codon frequencies for threonine and serine. The relative adaptiveness $w(c)$ of a codon c is the ratio of the frequency of c to the frequency of the most frequent codon encoding the same amino acid (white bar), and its value is shown to the right of the bars. Bottom, a weighted mRNA DFA encodes the CAI of each candidate in the total weight of its corresponding path by using $-\log[w(c)]$ (the cost of choosing codon c) as edge weights (Methods, 'Optimization objectives'). This weighted DFA is used as input to lattice parsing for joint optimization between stability and codon optimality.

lower than the optimal CAIs ($\lambda = \infty$). Also, our optimally stable designs have mostly double-stranded secondary structures (Fig. 3d), which are predicted to be much less prone to degradation[5]. By varying λ from 0 to ∞, LinearDesign computes the feasibility limit (optimal boundary) of the mRNA design space (the blue curves in Fig. 3b,c; see Extended Data Fig. 4 for λ in $(-\infty, 0]$). Furthermore, when the codon bias prefers AU-rich codons (such as in yeast), codon optimization actually worsens the stability (Extended Data Fig. 4b).

Results for COVID-19 mRNA vaccines

We examined eight mRNA sequences for the SARS-CoV-2 spike protein in this study. Seven sequences (sequences A–G) were designed with the LinearDesign algorithm as suboptimal molecules (with beam search[21,25]). They were widely distributed in the low-MFE design space (the region where MFE ≤ −1,400 kcal mol^{-1} in Fig. 4a), which is unreachable with a conventional codon optimizing algorithm. To obtain a better understanding of the biological effects of MFE and CAI, we designed these mRNA sequences to have almost identical values in either MFE (B and C have similar MFEs, while D, E and F have similar MFEs) or CAI (A, C and F have similar CAIs, B and E have similar CAIs and D, G and H have similar CAIs). The eighth mRNA sequence (sequence H) was designed with OptimumGene, a widely used codon-optimization tool, as a benchmark. This benchmark sequence has been used in a COVID-19 mRNA vaccine that elicited high immunogenicity in two animal models[26] and entered a phase I clinical trial in China (co-developed with the Chinese Center for Disease Control and Prevention; Chinese Clinical Trial Registry: CTR20210542). All of these mRNA sequences encode the

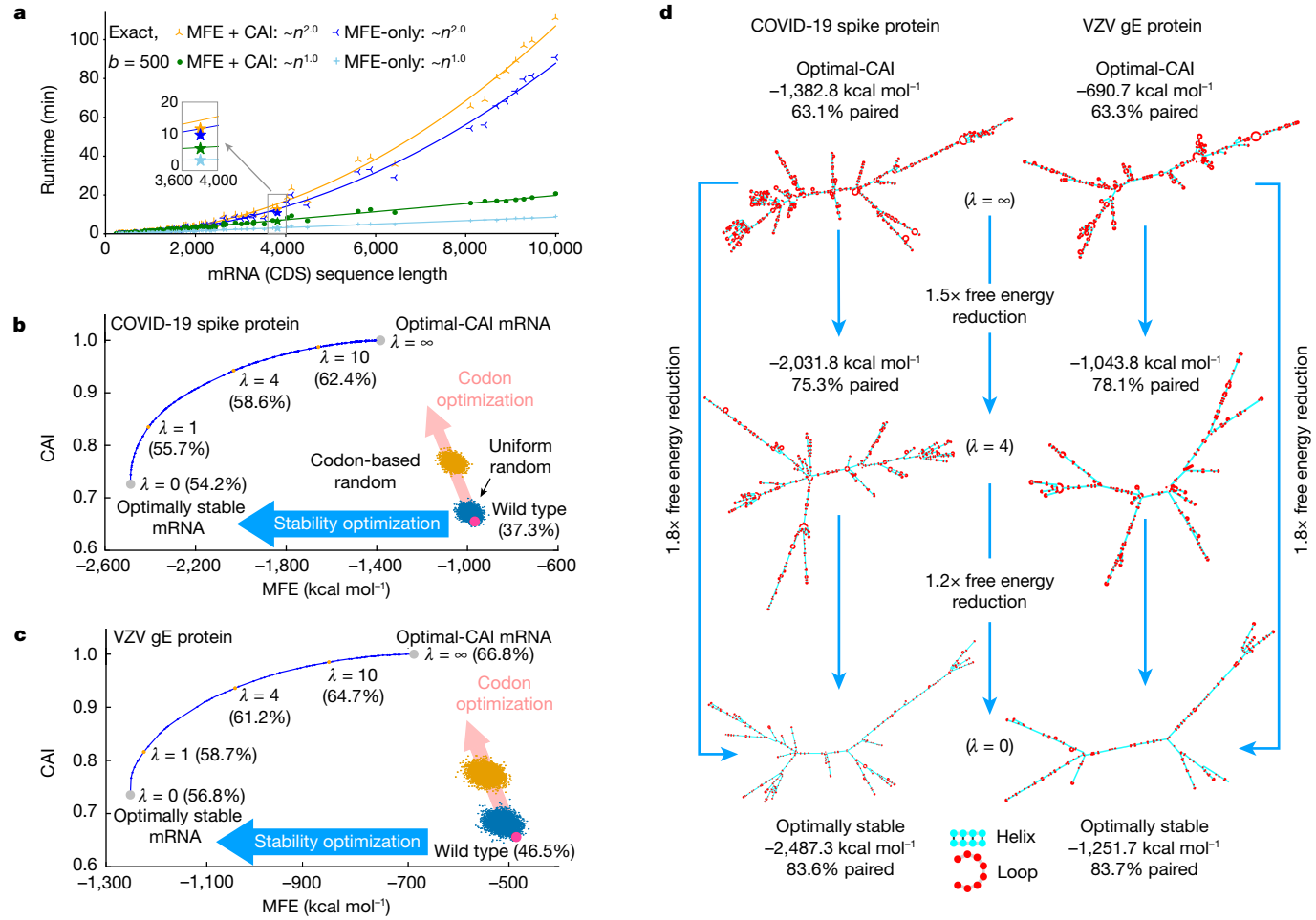


Fig. 3 | Computational characteristics of the LinearDesign algorithm. **a**, Runtime analysis of mRNA design for proteins in UniProt (Supplementary Table 1). Overall, our exact search scales quadratically with sequence length (Supplementary Figs. 7 and 8), and our MFE + CAI mode (with $\lambda = 4$) is about 15% slower than the MFE-only version. Moreover, beam search ($b = 500$) significantly speeds up the design of long sequences, with minor search errors (Supplementary Fig. 9). **b**,**c**, Two-dimensional (MFE–CAI) visualizations of designs for the SARS-CoV-2 spike (**b**) and VZV gE (**c**) proteins, respectively (both using human codon preference). The blue curves form the feasibility limit (optimal boundary), by varying λ from 0 to ∞ (see Extended Data Fig. 4 for λ of $(-\infty, 0]$). The GC percentage is shown in parentheses. The human genome favours GC-rich codons; therefore, codon optimization (pink arrows) also improves stability, but only marginally, as the two optimization directions (codon versus stability) are largely orthogonal. By contrast, with an AU-rich codon preference (such as in yeast), codon optimization decreases stability (Extended Data Fig. 4b). **d**, Secondary structures of the mRNA designs for SARS-CoV-2 spike and VZV gE protein. The optimal-CAI designs (top, $\lambda = \infty$) are largely single-stranded (around 60% base-paired), whereas the optimally stable designs (bottom, $\lambda = 0$) are mostly double-stranded (around 80% base-paired). We also show intermediate designs (centre, $\lambda = 4$) with a balance of stability and CAI.

same amino acid sequence of full-length wild-type SARS-CoV-2 spike protein, use natural unmodified nucleotides, and share the same 5′- and 3′-UTRs (see Supplementary Information for sequences).

Considering the potential negative effect on translation efficiency caused by a structured 5′-leader region[5], we did not include the first 5 amino acids when running LinearDesign, and instead used a heuristic to select the first 15 nucleotides. It has also been suggested that long helices may elicit unwanted innate immune responses[27], so we avoided them in our designs. This also explains why we did not study the lowest-MFE candidates (those closest to the optimal boundary—the blue curve in Fig. 4a), which usually contain long stems. See Methods, 'Additional design constraints' for details.

Besides coding region design, UTR structure is also crucial for translation[28] and UTR engineering has a profound effect on protein expression[3]. Although LinearDesign does not address UTR optimization per se, its designed mRNA molecules—as they are more structured than solely codon-optimized ones—form fewer base pairs with and thus interfere less with the structures of widely used UTRs (Extended Data Table 1). This was confirmed by a different pair of UTRs in our VZV mRNA vaccine experiments (Extended Data Table 2) leading to improved protein expression and immune responses (Fig. 5). This evidence suggests that LinearDesign is likely to remain effective independent of the choice of UTRs, which is also consistent with a recent study[29] in which LinearDesign-generated sequences with three different UTRs exhibited stronger in vitro protein expression over all benchmark sequences (see figure 4a in ref. 29); see Methods, 'Related work' for details.

In-solution structure compactness and chemical stability

We then studied the structural compactness of mRNA molecules, which is hypothesized to be correlated with the folding free-energy change. An mRNA molecule with a lower MFE tends to contain more secondary structures, exhibit a more compact shape and have a smaller hydrodynamic size, resulting in a higher electrophoretic mobility. We loaded mRNA samples onto a non-denaturing agarose gel and found that RNA

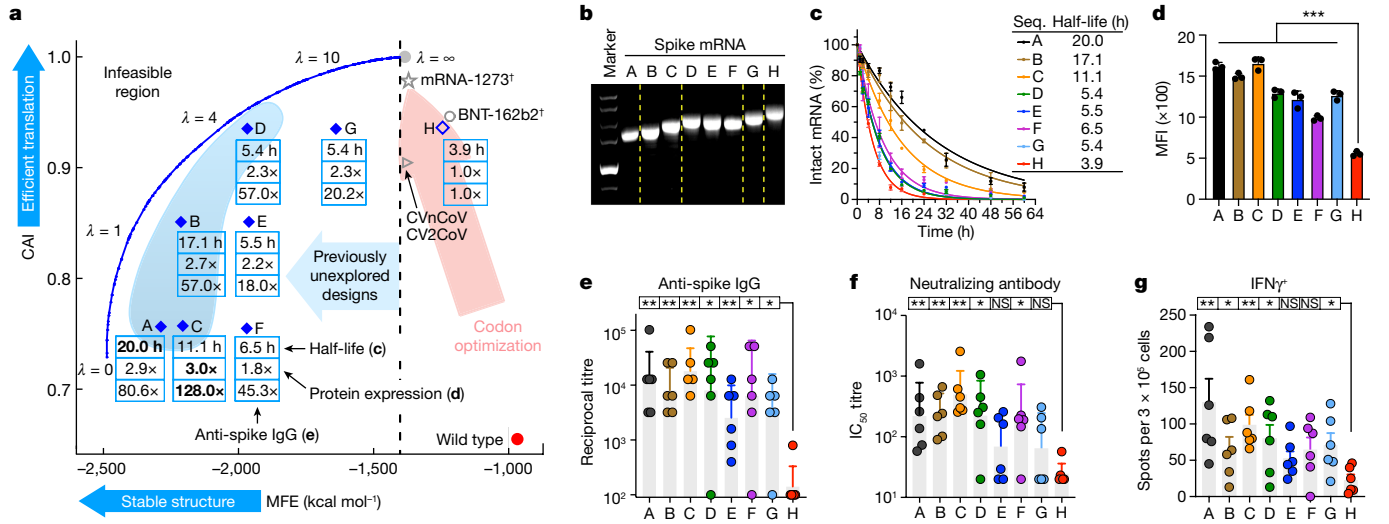


Fig. 4 | Experimental evaluation of LinearDesign-generated mRNA sequences encoding SARS-CoV-2 spike protein. a, Summary of chemical stability of and protein expression from spike mRNA designs A–G and the corresponding immune response (induction of anti-spike IgG) in mice compared to the codon-optimized benchmark H. The vaccines of mRNA-1273 and BNT-162b2 are annotated with daggers, because they use modified nucleotides, but their MFEs here are calculated with the standard energy model. **b**, Non-denaturing agarose gel characterization of mRNA, showing the correlation of gel mobility with minimum free energy. For gel source data, see Supplementary Fig. 1a. **c**, Chemical stability of mRNAs upon incubation in 10 mM Mg^{2+} buffer at 37 °C. Data are from three independent experiments. Seq., sequence. **d**, Protein expression levels from mRNAs 48 h after transfection into HEK293 cells, as determined by flow cytometry. Mean fluorescence intensity (MFI) values are derived from three independent experiments. Kruskal–Wallis ANOVA with Dunn's multiple comparisons with the H group. **e**–**g**, C57BL/6 mice ($n = 6$) were immunized intramuscularly with two doses of formulated mRNA with a two-week interval. **e**, End-point titre of anti-spike IgG. **f**, Levels of neutralizing antibodies against wild-type SARS-CoV-2. IC_{50}, half-maximal inhibitory concentration. **g**, Frequencies of IFNγ-secreting T cells, measured by enzyme-linked immunospot (ELISpot) assay. Two-tailed Mann–Whitney U test. Data are mean ± s.d. (**c**,**d**), geometric mean ± geometric s.d. (**e**,**f**) or mean ± s.e.m. (**g**). $^*P < 0.05$, $^{**}P < 0.01$, $^{***}P < 0.001$. NS, not significant. See Extended Data Figs. 5–7 and Supplementary Figs. 10 and 12 for extra experimental results and predicted secondary structures, and Supplementary Table 2 for detailed computational and experimental data.

mobility rates correlated well with the calculated MFEs for sequences A–H (Fig. 4b) despite the sequences having similar molecular weights. Sequence A, with the lowest MFE, exhibited the highest mobility, followed by other sequences in order of their MFEs. Sequence H, which has the highest MFE value, was the least mobile. These data demonstrated the validity of the MFE calculation executed by LinearDesign.

To evaluate the chemical stability of mRNAs, we incubated the mRNAs in buffers containing 10 mM (Fig. 4c) or 20 mM (Extended Data Fig. 5) Mg^{2+} at 37 °C, and assessed RNA integrity following incubation, similar to previous work[29]. Sequences A–H showed distinct degradation rates that correlated well with their MFEs (Fig. 4c and Extended Data Fig. 5). Sequence A, which has the lowest MFE, showed the slowest degradation rate, with a half-life ($T_{1/2}$) of 20.0 and 12.6 h in 10 and 20 mM Mg^{2+} buffers, respectively (Fig. 4c and Extended Data Fig. 5). By contrast, sequence H, which has the highest MFE, degraded the fastest with $T_{1/2}$ of 3.9 and 3.3 h in 10 and 20 mM Mg^{2+} buffers, respectively. These results support the idea that low-MFE designs are more resistant to in-solution degradation, a favourable trait for biological applications.

Cellular protein expression

For vaccines, sufficient antigen expression is a key determinant for eliciting effective immune responses. We thus evaluated protein expression of the designed mRNAs. Sequences A–H were translated efficiently into spike protein following transfection in HEK293 cells. Of note, all seven mRNAs generated by LinearDesign (sequences A–G) showed remarkably higher protein expression levels than benchmark sequence H (Fig. 4d and Supplementary Fig. 12). Sequences D and G (with CAIs almost identical to H, but lower MFEs) expressed 2.3-fold higher protein levels than sequence H, and sequence A, with the lowest MFE, showed 2.9-fold higher expression. Collectively, our results are consistent with those of Mauger et al.[5], which show that low MFE and high CAI synergize to improve protein expression; we were able to test this hypothesis using mRNA molecules with much lower MFE values, thanks to the ability of LinearDesign to explore the previously unreachable design space.

In vivo immunogenicity

We further tested whether these designs could endow increased immunogenicity in vivo. We inoculated mRNA sequences A–H into mice using a lipid-based formulation[30], and evaluated their humoral and cellular immune responses. For each mRNA sequence, C57BL/6 mice were inoculated intramuscularly with two doses of the vaccines with an interval of two weeks. Levels of anti-spike IgG, neutralizing antibodies and spike-specific interferon-γ (IFNγ)-secreting T cells were assessed. All mRNA molecules from LinearDesign were able to elicit robust antibody responses. By contrast, sequence H mRNA showed very limited ability to induce antibodies (Fig. 4e,f). Similar results were also observed on the antigen-specific T cell response, where a robust T helper 1-biased T cell response was induced only by the LinearDesign mRNAs (Fig. 4g). Sequences A–D, which are closer to the optimal boundary (blue shaded region in Fig. 4a), elicited a 57 to 128× increase in anti-spike IgG antibody titres and a 9 to 20× increase in neutralizing antibody titres over those elicited by the benchmark sequence H.

Since BNT-162b2 from BioNTech-Pfizer is the most widely adopted COVID-19 mRNA vaccine, we compared it with the LinearDesign mRNAs. For this head-to-head comparison, our BNT sequence is almost identical to the sequence of BNT-162b2, but with three changes: (a) the two stabilizing proline mutations[31] in BNT-162b2 converted back to the wild-type sequence, (b) BNT uses the same 5′- and 3′-UTRs as in sequences A–H, and (c) it uses natural unmodified nucleotides as in sequences A-H. Four mRNA sequences—A, C, H and BNT—were included in the study. A and C showed a markedly lower in-solution degradation rate and significantly higher protein expression in HEK293 cells than BNT (Extended Data Fig. 6). Note that BNT and H have very similar

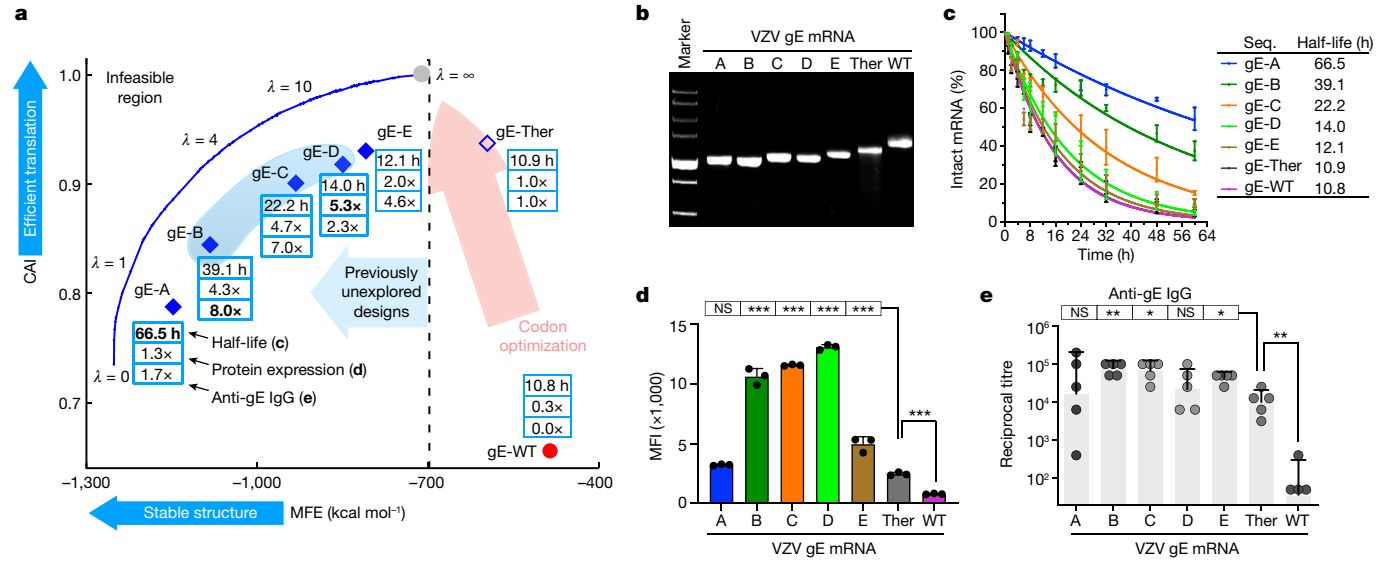


Fig. 5 | Experimental evaluation of LinearDesign-generated mRNAs encoding VZV gE protein. a, Summary of chemical stability of and protein expression from VZV gE mRNA designs and the corresponding immune response (induction of anti-gE IgG) in mice. The 'sweet spot' region is highlighted with light blue shading. **b**, Non-denaturing agarose gel characterization of mRNA showing the correlation of gel mobility with minimum free energy; for gel source data, see Supplementary Fig. 1b. **c**, Chemical stability of mRNAs upon incubation in 10 mM Mg^{2+} buffer at 37 °C. Data are from three independent experiments. **d**, Protein expression levels from mRNAs 48 h after transfection into HEK293 cells, as determined by flow cytometry. MFI values are derived from three independent experiments. Kruskal–Wallis ANOVA with Dunn's multiple comparisons with the gE-Ther group. **e**, C57BL/6 mice ($n = 5$) were immunized intramuscularly with two doses of formulated mRNA with a two-week interval. End-point titre of anti-gE IgG is shown. Two-tailed Mann–Whitney U test. Data are mean ± s.d. (**c**,**d**) or geometric mean ± geometric s.d. (**e**). See Extended Data Fig. 8 for extra experimental results, Supplementary Fig. 11 for predicted secondary structures and Supplementary Table 3 for detailed computational and experimental data.

MFEs, CAIs (Fig. 4a) and half-lives. Moreover, A and C were able to elicit significantly higher levels of anti-spike IgG and neutralizing antibodies than H and BNT (Extended Data Fig. 7). Collectively, these data lead us to speculate that LinearDesign-optimized mRNA molecules are more stable in vivo, which leads to improved protein expression and enhanced immunogenicity.

Results for VZV mRNA vaccines

To further evaluate the generalizability of LinearDesign, we applied the algorithm to the design of a mRNA vaccine for VZV. Vaccination against VZV is considered an effective approach to reduce the risk of shingles[32]. Using the same strategy as for spike mRNA design (Fig. 4a), we generated five mRNA sequences encoding the full-length VZV gE protein (gE-A to gE-E). These sequences are widely distributed in the previously unexplored high-thermostability region (Fig. 5a). These sequences were benchmarked to the gE-Ther sequence, which we designed with the widely used codon-optimization tool GeneOptimizer[33]. These mRNAs, including wild-type gE mRNA (gE-WT), shared the same encoded amino acid sequence and 5′ and 3′ UTRs (the sequences are provided in Supplementary Information). In line with the spike mRNA data (Fig. 4b), gE-A mRNA, which has the lowest MFE, showed the greatest mobility in a non-denaturing gel (Fig. 5b) and markedly slower degradation rates with a $T_{1/2}$ of 66.5 h in 10 mM (Fig. 5c) and 50.7 h in 20 mM (Extended Data Fig. 8a) Mg^{2+} buffers, indicating a high chemical stability correlated with the compactness of molecules. By contrast, gE-Ther showed a $T_{1/2}$ of 10.9 h in 10 mM and 5.9 h in 20 mM Mg^{2+} buffers. We also observed that gE mRNA molecules were more stable than spike mRNAs owing to their shorter length[34]. In addition, protein expression from most of the LinearDesign-generated mRNAs (gE-B to gE-E) was significantly higher than for gE-Ther and gE-WT in HEK293 cells 48 h (Fig. 5d) and 24 h (Extended Data Fig. 8b) after transfection. However, the best-performing mRNA molecules were gE-B, gE-C and gE-D. They outperformed gE-A, which has the lowest CAI, and gE-E, which has the highest MFE. This emphasizes the importance of jointly optimizing CAI and MFE. The most highly expressed molecules were those whose CAI and MFE were both in the favourable region (light blue shaded area; Fig. 5a). Finally, we evaluated the immune respopnse elicited by VZV mRNA vaccines in C57BL/6 mice. LinearDesign mRNA molecules (gE-B, gE-C and gE-E) induced significantly higher levels of anti-gE IgG than gE-Ther or gE-WT (Fig. 5e).

Discussion

An effective mRNA design strategy is of utmost importance for the development of mRNA vaccines, which have shown great promise against the COVID-19 pandemic. However, this task remains challenging owing to the prohibitively large search space. Here we present a simple solution by reducing the mRNA design problem to the classical problem of lattice parsing used in computational linguistics. This work resulted in an efficient algorithm that can design an optimal mRNA encoding the SARS-CoV-2 spike protein in 11 min. It can also jointly optimize stability and codon usage, which has been shown to be crucial for mRNA design. This approach is one of several recent fruitful exchanges between linguistics and biology[35,36].

Here we have comprehensively characterized mRNA sequences generated by LinearDesign and demonstrated their superiority over the commonly used codon-optimization benchmark using two viral antigens across three attributes that are critical for vaccine performance: chemical stability, protein translation and in vivo immunogenicity. In particular, our designs for mRNA encoding the SARS-CoV-2 spike protein showed an increase of up to 128-fold in binding antibody levels over the codon-optimization benchmark. Our VZV gE mRNA designs—using a different UTR pair—also showed substantial improvements over the benchmark. These results indicate the robustness of LinearDesign in optimizing the coding region independently of UTR pairs. Indeed,

coding region design and UTR engineering[3] are complementary and could be combined in future work. It is worth noting that our designed mRNAs did not use chemical modification which is widely believed to be critical to the recent success of mRNA vaccines[1,2,10,37,38], yet still showed high levels of stability, translation efficiency and immunogenicity, with the additional advantage of a lower manufacturing cost. The LinearDesign approach is likely to complement chemical modification and can be easily adapted to modified nucleotides once the corresponding energy model is available. Our work has only considered stability and codon usage but, owing to the generalizability of the lattice representation, could also be adapted to optimize other parameters relevant to mRNA design. By opening up the previously inaccessible region of highly stable and efficient sequences, this approach provides a timely and promising tool for mRNA vaccine development that is likely to have a key role in future pandemics. It is also a principled method for molecule design in the field of mRNA medicines generally, and can be used for all therapeutic proteins including monoclonal antibodies and anti-cancer drugs.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41586-023-06127-z.

1. Baden, L. R. et al. Efficacy and safety of the mRNA-1273 SARS-CoV-2 vaccine. *N. Engl. J. Med.* **384**, 403–416 (2021).
2. Polack, F. P. et al. Safety and efficacy of the BNT162b2 mRNA Covid-19 vaccine. *N. Engl. J. Med.* **383**, 2603–2615 (2020).
3. Gebre, M. S. et al. Optimization of non-coding regions for a non-modified mRNA COVID-19 vaccine. *Nature* **601**, 410–414 (2022).
4. Crommelin, D. J., Anchordoquy, T. J., Volkin, D. B., Jiskoot, W. & Mastrobattista, E. Addressing the cold reality of mRNA vaccine stability. *J. Pharm. Sci.* **110**, 997–1001 (2021).
5. Mauger, D. M. et al. mRNA structure regulates protein expression through changes in functional half-life. *Proc. Natl Acad. Sci. USA* **116**, 24075–24083 (2019).
6. Hall, K. B. *Best-First Word-Lattice Parsing: Techniques for Integrated Syntactic Language Modeling*. PhD thesis, Brown Univ. (2005).
7. Schlake, T. et al. mRNA: a novel avenue to antibody therapy? *Mol. Ther.* **27**, 773–784 (2019).
8. Reinhard, K. et al. An RNA vaccine drives expansion and efficacy of claudin-CAR-T cells against solid tumors. *Science* **367**, 446–453 (2020).
9. Wolff, J. A. et al. Direct gene transfer into mouse muscle in vivo. *Science* **247**, 1465–1468 (1990).
10. Pardi, N., Hogan, M. J., Porter, F. W. & Weissman, D. mRNA vaccines—a new era in vaccinology. *Nat. Rev. Drug Discov.* **17**, 261–279 (2018).
11. Mauro, V. P. & Chappell, S. A. A critical analysis of codon optimization in human therapeutics. *Trends Mol. Med.* **20**, 604–13 (2014).
12. Gustafsson, C., Govindarajan, S. & Minshull, J. Codon bias and heterologous protein expression. *Trends Biotechnol.* **22**, 346–353 (2004).
13. Nabiyouni, M., Prakash, A. & Fedorov, A. Vertebrate codon bias indicates a highly GC-rich ancestral genome. *Gene* **519**, 113–119 (2013).
14. Sahin, U., Karikó, K. & Türeci, Ö. mRNA-based therapeutics—developing a new class of drugs. *Nat. Rev. Drug Discov.* **13**, 759–780 (2014).
15. Mathews, D. H., Sabina, J., Zuker, M. & Turner, D. H. Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *J. Mol. Biol.* **288**, 911–940 (1999).
16. Mathews, D. H. et al. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc. Natl Acad. Sci. USA* **101**, 7287–7292 (2004).
17. Sharp, P. M. & Li, W. H. The codon adaptation index–a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic Acids Res.* **15**, 1281–1295 (1987).
18. Durbin, R., Eddy, S. R., Krogh, A. & Mitchison G. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids* (Cambridge Univ. Press, 1998).
19. Bar-Hillel, Y., Perles, M. & Shamir, E. On formal properties of simple phrase structure grammars. *Zeitschrift für Phonetik, Sprachwissenschaft und Kommunikationsforschung* **14**, 143–172 (1961).
20. Nederhof, M. J. & Satta, G. Probabilistic parsing as intersection. In *Proc. 8th International Conference on Parsing Technologies* 137–148 (2003).
21. Huang, L. et al. LinearFold: linear-time approximate RNA folding by 5′-to-3′ dynamic programming and beam search. *Bioinformatics* **35**, i295–i304 (2019).
22. Cohen, B. & Skiena, S. Natural selection and algorithmic design of mRNA. *J. Comput. Biol.* **10**, 419–432 (2003).
23. Terai, G., Kamegai, S. & Asai, K. CDSfold: an algorithm for designing a protein-coding sequence with the most stable secondary structure. *Bioinformatics* **32**, 828–834 (2016).
24. Consertium, U. UniProt: a hub for protein information. *Nucleic Acids Res.* **42**, D204–D12 (2005).
25. Huang, L. & Sagae, K. Dynamic programming for linear-time incremental parsing. In *Proc. 48th Annual Meeting of the Association for Computational Linguistics* 1077–1086 (Association for Computational Linguistics, 2010).
26. Yang, R. et al. A core-shell structured COVID-19 mRNA vaccine with favorable biodistribution pattern and promising immunity. *Signal Transduct. Target. Ther.* **6**, 213 (2021).
27. Liu, L. et al. Structural basis of Toll-like receptor 3 signaling with double-stranded RNA. *Science* **320**, 379–381 (2008).
28. Mignone, F., Gissi, C., Liuni, S. & Pesole, G. Untranslated regions of mRNAs. *Genome Biol.* **3**, reviews0004.1 (2002).
29. Leppek, K. et al. Combinatorial optimization of mRNA structure, stability, and translation for RNA-based therapeutics. *Nat. Commun.* **13**, 1536 (2022).
30. Rana, M. M. Polymer-based nano-therapies to combat COVID-19 related respiratory injury: progress, prospects, and challenges. *J. Biomater. Sci. Polym. Ed.* **32**, 1219–1249 (2021).
31. Wrapp, D. et al. Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science* **367**, 1260–1263 (2020).
32. Cunningham, A. L. et al. Efficacy of the herpes zoster subunit vaccine in adults 70 years of age or older. *N. Engl. J. Med.* **375**, 1019–1032 (2016).
33. Raab, D., Graf, M., Notka, F., Schödl, T. & Wagner, R. The GeneOptimizer algorithm: using a sliding window approach to cope with the vast sequence space in multiparameter DNA sequence optimization. *Syst. Synth. Biol.* **4**, 215–225 (2010).
34. Wayment-Steele, H. K. et al. Theoretical basis for stabilizing messenger RNA through secondary structure design. *Nucleic Acids Res.* **49**, 10604–10617 (2021).
35. Hie, B., Zhong, E. D., Berger, B. & Bryson, B. Learning the language of viral evolution and escape. *Science* **371**, 284–288 (2021).
36. Madani A., et al. Large language models generate functional protein sequences across diverse families. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-022-01618-2 (2023).
37. Karikó, K., Buckstein, M., Ni, H. & Weissman, D. Suppression of RNA recognition by Toll-like receptors: the impact of nucleoside modification and the evolutionary origin of RNA. *Immunity* **23**, 165–175 (2005).
38. Karikó, K. et al. Incorporation of pseudouridine into mRNA yields superior nonimmunogenic vector with increased translational capacity and biological stability. *Mol. Ther.* **16**, 1833–1840 (2008).

Methods

The LinearDesign algorithm

Optimization objectives. There are two objectives in mRNA design: stability and codon optimality. The optimal-stability mRNA design problem can be formalized as follows. Given a protein sequence $\mathbf{p} = p_0 \ldots p_{|p|-1}$ where each p_i is an amino acid residue, we find the optimal mRNA sequence $\mathbf{r}^*(\mathbf{p})$ that has the lowest MFE among all possible mRNA sequences encoding that protein:

$$\mathbf{r}^*(\mathbf{p}) = \operatorname{argmin}_{\mathbf{r} \in \mathrm{mRNA}(\mathbf{p})} \mathrm{MFE}(\mathbf{r}) \tag{1}$$

$$\mathrm{MFE}(\mathbf{r}) = \min_{\mathbf{s} \in \mathrm{structures}(\mathbf{r})} \Delta G^\circ(\mathbf{r}, \mathbf{s}) \tag{2}$$

where $\mathrm{mRNA}(\mathbf{p}) = \{\mathbf{r} | \mathrm{protein}(\mathbf{r}) = \mathbf{p}\}$ is the set of candidate mRNA sequences, $\mathrm{structures}(\mathbf{r})$ is the set of all possible secondary structures for mRNA sequence $\mathbf{r}$, and $\Delta G^\circ(\mathbf{r}, \mathbf{s})$ is the free-energy change of structure $\mathbf{s}$ for mRNA $\mathbf{r}$ according to an energy model. This is clearly a double minimization objective involving the per-sequence minimization over all of its possible structures (that is, RNA folding; equation (2)), which has well-known dynamic programming solutions, and the global minimization over all sequences (that is, optimal mRNA design; equation (1)) which we will solve using lattice parsing (see 'SCFG, lattice parsing and intersection').

Next, we integrate codon optimality by adding CAI[17], defined as the geometric mean of the codon optimality of each codon in the mRNA $\mathbf{r}$:

$$\mathrm{CAI}(\mathbf{r}) = \sqrt[\frac{|\mathbf{r}|}{3}]{\prod_{0 \le i < \frac{|\mathbf{r}|}{3}} w(\mathrm{codon}(\mathbf{r}, i))} \tag{3}$$

where $\mathrm{codon}(\mathbf{r}, i) = r_{3i} r_{3i+1} r_{3i+2}$ is the ith triplet codon in $\mathbf{r}$, and $w(c)$ is the relative adaptiveness of codon c, defined as the frequency of c divided by the frequency of its most frequent synonymous codon ($0 \le w(c) \le 1$). Because CAI is always between 0 and 1 but MFE is generally proportional to the mRNA sequence length, we scale CAI by the number of codons and use a hyper-parameter λ to balance MFE and CAI ($\lambda = 0$ being purely MFE), and define a novel joint objective:

$$\mathrm{MFECAI}_\lambda(\mathbf{r}) = \mathrm{MFE}(\mathbf{r}) - \frac{|\mathbf{r}|}{3} \lambda \log \mathrm{CAI}(\mathbf{r}) \tag{4}$$

which can be simplified by expanding CAI:

$$\begin{aligned} \mathrm{MFECAI}_\lambda(\mathbf{r}) &= \mathrm{MFE}(\mathbf{r}) - \frac{|\mathbf{r}|}{3} \lambda \log \sqrt[\frac{|\mathbf{r}|}{3}]{\prod_{0 \le i < \frac{|\mathbf{r}|}{3}} w(\mathrm{codon}(\mathbf{r}, i))} \\ &= \mathrm{MFE}(\mathbf{r}) - \lambda \sum_{0 \le i < \frac{|\mathbf{r}|}{3}} \log w(\mathrm{codon}(\mathbf{r}, i)) \end{aligned} \tag{5}$$

This joint objective is basically MFE plus (a scaled) sum of the negative logarithm of each codon's relative adaptiveness. Now the joint optimization can be defined as:

$$\begin{aligned} \mathbf{r}^*_\lambda(\mathbf{p}) &= \operatorname{argmin}_{\mathbf{r} \in \mathrm{mRNA}(\mathbf{p})} \mathrm{MFECAI}_\lambda(\mathbf{r}) \\ &= \operatorname{argmin}_{\mathbf{r} \in \mathrm{mRNA}(\mathbf{p})} \left(\mathrm{MFE}(\mathbf{r}) - \lambda \sum_{0 \le i < \frac{|\mathbf{r}|}{3}} \log w(\mathrm{codon}(\mathbf{r}, i))\right) \end{aligned} \tag{6}$$

See Fig. 2d for examples of relative adaptiveness calculation.

DFA representations for codons and mRNA candidate sequences. Informally, a DFA is a directed graph with labelled edges and distinct start and end states. For our purpose each edge is labelled by a nucleotide, so that for each codon DFA, each start-to-end path represents a triplet codon; see Fig. 2a and Extended Data Fig. 1c for examples. Formally, a DFA is a 5-tuple $\langle Q, \Sigma, \delta, q_0, F \rangle$, where Q is the set of states, Σ is the alphabet (here $\Sigma = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}\}$), q_0 is the start state (always (0,0) in this work), F is the set of end states (in this work the end state is unique—that is, $F = \{(3,0)\}$), and δ is the transition function that takes a state q and a symbol $a \in \Sigma$ and returns the next state q'—that is, $\delta(q, a) = q'$ encodes a labelled edge $q \xrightarrow{a} q'$.

After building DFAs for each amino acid, we can concatenate them (concatenation is indicated by $\circ$ below) into a single DFA $D(\mathbf{p})$ for a protein sequence $\mathbf{p}$, which represents all possible mRNA sequences that translate into that protein

$$D(\mathbf{p}) = D(p_0) \circ D(p_1) \circ \cdots \circ D(p_{|\mathbf{p}|-1}) \circ D(\mathrm{STOP})$$

by stitching the end state of each DFA with the start state of the next. See Extended Data Fig. 1d for examples. The new end state of the mRNA DFA is $(3|\mathbf{p}| + 3, 0)$.

We also define $\mathrm{out\_edges}(q)$ to be the set of outgoing edges from state q, and $\mathrm{in\_edges}(q)$ to be the set of incoming edges (which will be used in the pseudocode; Supplementary Figs. 2 and 3):

$$\mathrm{out\_edges}(q) = \{q \xrightarrow{a} q' | \delta(q, a) = q'\}$$
$$\mathrm{in\_edges}(q) = \{q' \xrightarrow{a} q | \delta(q', a) = q\}$$

For the mRNA DFA in Extended Data Fig. 1d, $\mathrm{out\_edges}((3,0)) = \{(3,0) \xrightarrow{\mathrm{U/G}} (4,0), (3,0) \xrightarrow{\mathrm{C}} (4,1)\}$ and $\mathrm{in\_edges}((9,0)) = \{(8,0) \xrightarrow{\mathrm{A}} (9,0), (8,0) \to (9,0), (8,1) \xrightarrow{\mathrm{A}} (9,0)\}$.

SCFG, lattice parsing and intersection. A SCFG is a context-free grammar in which each rule is augmented with a weight. More formally, an SCFG is a 4-tuple $\langle N, \Sigma, P, S \rangle$ where N is the set of non-terminals, Σ is the set of terminals (identical to the alphabet in the DFA, in this case $\Sigma = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}\}$), P is the set of weight-associated context-free writing rules, and $S \in N$ is the start symbol. Each rule in P has the form $A \xrightarrow{w} (N \cup \Sigma)^*$ where $A \in N$ is a non-terminal that can be rewritten according to this rule into a sequence of non-terminals and terminals (the star $*$ means repeating zero or more times) and $w \in \mathbb{R}$ is the weight associated with this rule.

SCFGs can be used to represent the RNA folding energy model[39]. The weight of a derivation (parse tree, or a secondary structure in this case) is the sum of weights of the productions used in that derivation. For example, for a very simple Nussinov–Jacobson-style model[40], which simplifies the energy model to individual base pairs, we can define this SCFG G as in Extended Data Fig. 1e, where each GC pair gets a score of −3, and each AU pair gets a score of −2. Thus, the standard RNA secondary structure prediction problem can be cast as a parsing problem: given the above SCFG G and an input RNA sequence, find the minimum-weight derivation in G that can generate the sequence. This can be solved by the classical CKY algorithm from computational linguistics[41–43].

The optimal-stability mRNA design problem is now a simple extension of the above single-sequence folding problem to the case of multiple inputs: instead of finding the minimum-free-energy structure (minimum-weight derivation) for a given sequence, we find the minimum-free-energy structure (and its corresponding sequence) among all possible structures for all possible sequences (Extended Data Fig. 1). This can be solved by lattice parsing on the DFA, which is a generalization of CKY from a single sequence to a DFA. Take the bifurcation rule $S \xrightarrow{N} NP$ for example. In CKY, if you have derived non-terminal N for span $[i, j]$, notated $i \xrightarrow{\Delta} j$, and if you have also derived $j \xrightarrow{\Delta} k$, you can combine the two spans—that is, $i \xrightarrow{\Delta} j \xrightarrow{\Delta} k$—and use the above rule to derive $i \xrightarrow{\Delta} k$. Similarly, in lattice parsing, if you have derived both $q_i \overset{N}{\rightsquigarrow} q_j$

(that is, there is a $q_i \rightsquigarrow q_j$ path that can be derived from N) and $q_j \overset{P}{\underset{S}{\Delta}} q_k$, you can combine them to a longer path $q_i \overset{N}{\underset{}{\Delta}} q_j \overset{P}{\underset{}{\Delta}} q_k$ and derive $q_i \overset{\Delta}{\underset{}{\rightsquigarrow}} q_k$ with the above rule. While the runtime for CKY scales $O(|G|\ n^3)$ where $|G|$ is the grammar constant (the number of rules) and n is the RNA sequence length, the runtime for lattice parsing similarly scales $O(|G|\,|D|^3)$ where $|D|$ is the number of states in the DFA. For mRNA design with the standard genetic code, $n \le |D| \le 2n$ because each position i has either one or two states ($(i, 0)$ and $(i, 1)$), so its time complexity is also actually identical to single-sequence folding, just with a larger constant. See 'Left-to-right dynamic programming' for details of this algorithm and Supplementary Figs. 2 and 3 for the pseudocode.

More formally, in theoretical computer science, lattice parsing with an CFG G on a DFA D is also known as the intersection between the languages of G and D (that is, the sets of sequences allowed by G and D), notated $L(G) \cap L(D)$, which was solved by the Bar-Hillel construction in 1961 (ref. 19). In order to adapt it to mRNA design, we need to extend this concept to the case of weighted (that is, stochastic) grammars and weighted DFAs (the latter is needed for CAI integration; see below). While the language $L(G)$ of CFG G is the set of sequences generated by G, the language of the SCFG for RNA folding free-energy model defines a mapping from each RNA sequence to its MFE—that is, $L_w(G) : \Sigma^* \longmapsto \mathbb{R}$. This can be written as a relation:

$$L_w(G) = \{\mathbf{r} \sim \mathrm{MFE}(\mathbf{r}) \mid \mathbf{r} \in \Sigma^*\}$$

And we also extend the language of a DFA to a trivial weighted language (which will facilitate the incorporation of CAI into DFA below):

$$L_w(D) = \{\mathbf{r} \sim 0 \mid \mathbf{r} \in L(D)\}$$

Next we extend the intersection from two sets to two weighted sets A and B:

$$A \bigcap_w B = \{\mathbf{r} \sim (w_1 + w_2) \mid \mathbf{r} \sim w_1 \in A, \mathbf{r} \sim w_2 \in B\}$$

Now we can show that optimal-stability mRNA design problem can be solved via weighted intersection between $L_w(G)$ and $L_w(D)$—that is, we can construct a new 'intersected' stochastic grammar G' that has the same weights (that is, energy model) as the original grammar but only generates sequences in the DFA:

$$L_w(G') = L_w(G) \bigcap_w L_w(D) = \{\mathbf{r} \sim \mathrm{MFE}(\mathbf{r}) \mid \mathbf{r} \in L(D)\}$$

Weighted DFA for CAI integration. As described in the main text and Fig. 2d, our novel joint optimization objective (equation (6)) factors the CAI of each mRNA candidate onto the relative adaptiveness of each of its codons, and thus can be easily incorporated into the DFA as edge weights. To do this we need to extend the definition of DFA to weighted DFA, where the transition function δ now returns a state and a weight—that is, $\delta(q, a) = (q', w)$ —which encodes a weighted label edge $q \overset{a:w}{\rightarrow} q'$. Now the set of outgoing and incoming edges are also updated to:

$$\mathrm{out\_edges}(q) = \{q \overset{a:w}{\rightarrow} q' | \delta(q, a) = (q', w)\}$$
$$\mathrm{in\_edges}(q) = \{q' \overset{a:w}{\rightarrow} q | \delta(q', a) = (q, w)\}$$

In this case, the weighted DFA defines a mapping from each candidate mRNA sequence to its negative logarithm of CAI scaled by the number of codons—that is, $L_w(D) : L(D) \mapsto \mathbb{R}$. More formally,

$$L_w(D) : \{\mathbf{r} \sim -\frac{|\mathbf{r}|}{3}\log\mathrm{CAI}(\mathbf{r}) | \ \mathbf{r} \in L(D)\}$$

Now the weighted intersection defined above can be extended to incorporate the hyper-parameter λ and derive the joint objective:

$$L_w^{\lambda}(G') = L_w(G) \bigcap_w^{\lambda} L_w(D) = \{\mathbf{r} \sim (\mathrm{MFE}(\mathbf{r}) - \lambda\frac{|\mathbf{r}|}{3}\log\mathrm{CAI}(\mathbf{r})) | \ \mathbf{r} \in L(D)\}$$

Bottom-up dynamic programming. Next, we describe how to implement the dynamic programming algorithm behind lattice parsing (or equivalently, intersection between the languages of a SCFG and a weighted DFA) to solve the joint optimization problem. For simplicity reasons, here we use bottom-up dynamic programming on a modified Nussinov–Jacobson energy model. Supplementary Fig. 2 gives the pseudocode for this simplified version. We first build up the mRNA DFA for the given protein, and initialize two hash tables, 'best' to store the best score of each state, and 'back' to store the best backpointer. For the base cases ($S \overset{0}{\rightarrow} N\ N\ N$) we set $\mathrm{best}[S, q_i, q_{i+3}] \leftarrow 0$ for optimal-stability design, and $\mathrm{best}[S, q_i, q_{i+3}] \leftarrow \mathrm{mincost}(q_i, q_{i+3}, \lambda)$ for the joint optimization where

$$\mathrm{mincost}(q_i, q_{i+3}, \lambda) \triangleq \min_{q_i \xrightarrow{a:w_1} q' \xrightarrow{b:w_2} q'' \xrightarrow{c:w_3} q_{i+3}} \lambda(w_1 + w_2 + w_3) \tag{7}$$

is the minimum (λ-scaled) cost of any $q_i \rightsquigarrow q_{i+3}$ path in the CAI-integrated DFA. Next, for each state (q_i, q_j) it goes through the pairing rule and bifurcation rules, and updates if a better score is found. After filling out the hash tables bottom-up, we can backtrace the best mRNA sequence stored with the backpointers. See Supplementary Fig. 3 for details of UPDATE and BACKTRACE functions.

Left-to-right dynamic programming. Inspired by our previous work, LinearFold[21], we further developed a left-to-right dynamic programming, which is equivalent to the above bottom-up version but explores the search space incrementally from left to right; see Supplementary Fig. 4 for the pseudocode. This left-to-right order also enables beam search[44], a classical pruning technique, to significantly narrow down the search space without sacrificing too much search quality. Our real system uses this left-to-right dynamic programming on the Turner nearest-neighbour free-energy model[15,16], and our thermodynamic parameters follow LinearFold and Vienna RNAfold[45], except for the dangling ends, which do not contribute stability in LinearDesign. Dangling ends refer to stabilizing interactions for multiloops and external loops[46], which require knowledge of the nucleotide sequence outside of the state (q_i, q_j). Though it could be integrated in LinearDesign, the implementation is more involved so we leave it to future work.

DFAs for other genetic codes, coding constraints and modified nucleotides. The DFA framework can also represent less common cases such as alternative genetic codes, modified nucleotides, and coding constraints. First, the DFA can encode non-standard genetic codes, such as alternative nuclear code for some yeast[47] and mitochondrial codes[48] (Extended Data Fig. 3a). Second, we may want to avoid some unwanted or rare codons (such as the amber stop codon) which is an easy change on the codon DFAs (Extended Data Fig. 3b), or certain adjacent codon pairs that modulate translation efficiency[49], which is beyond the scope of single codon DFAs but easy on the mRNA DFA (Extended Data Fig. 3c). Similarly, we may want to disallow certain restriction enzyme recognition sites, which span across multiple codons (Supplementary Fig. 5). Finally, chemically modified nucleotides such as pseudouridine (Ψ) have been widely used in mRNA vaccines[38], which can also be incorporated in the DFA (Extended Data Fig. 3d).

Related work. Here we first discuss the advantages of our algorithm over previous work, and then discuss a recent work[29] that uses LinearDesign in experimental screening.

Two previous studies[22,23] also tackled the problem of optimal-stability mRNA design (that is, our objective 1) via dynamic programming, but their algorithms are complicated, not generalizable and less efficient. By contrast, the stability-only version of our work reduced the mRNA design problem to the classical computational linguistics problem of lattice parsing, resulting in a much simpler and more efficient algorithm that is vastly different from the specifically-designed algorithms such as the one described in Cohen et al.[22] and CDSfold[23]. More importantly, our work further solves the harder and practically more important problem of joint optimization between stability and codon optimality, which subsumes the stability-only objective as a special case. Here we comprehensively compare our work to the previous ones in the following seven aspects.

Lattice representation of the design space. Our work is the first to use automata theory to compactly and conveniently represent the exponentially large mRNA design space. By contrast, Cohen et al. and CDSfold extend the standard Zuker algorithm with the consideration of amino acid constraints, and they do not have any graph-theoretic or formal representation of the design space. To handle the nucleotide dependencies of the first and third positions in the codons of leucine and arginine, CDSfold introduces the 'extended nucleotides', which classify the same nucleotide at the second position with different notations regarding the dependency. See Supplementary Fig. 6 for the lattice representation of leucine in our work as an example, and the extended nucleotides of leucine in CDSfold as a comparison. More importantly, our lattice representation is able to integrate (the logarithm) of CAI for a joint optimization of stability and codon optimality, and is general for arbitrary genetic code; see the details in later paragraphs.

Lattice parsing. Based on our DFA representation, we further reduce the mRNA design problem to the classical computational linguistics problem of lattice parsing, which aims to find the most grammatical sentence among exponentially many alternatives. This problem was solved by Bar-Hillel et al. in 1961 (ref. 19). Therefore, instead of inventing a new algorithm, we simply adapt the classical lattice parsing to mRNA design using our algorithm of LinearDesign. Note that the single-sequence folding is a special case of our algorithm where the lattice is a single chain.

Efficiency. More interestingly, our simple adapted algorithm reduces the constant factor of the cubic-time bifurcation rule that dominates the runtime of mRNA design, leading to better efficiency over previous work such as CDSfold. Supplementary Fig. 7b illustrates the space and time complexity under the classical Nussinov energy model.

The single-sequence RNA folding defines a span (i,j) as an item, where i and j are indices in the RNA sequence. For a sequence with n nucleotides, during dynamic programming, at most n^3 items are generated for the bifurcation rule $S \rightarrow S\,P$; space-wise, at most n^2 items are stored.

Extending RNA folding to lattice parsing, our work defines each item as (q_i, q_j), where q_i and q_j are the nodes in the lattice: $q_i \in \{(i,0), (i,1)\}$ and $q_j \in \{(j,0), (j,1)\}$. Since there are at most two nodes at each position, the number of items stored is at most $4n^2$. For the bifurcation rule $S \rightarrow S\,P$, items (q_i, q_k) and (q_k, q_j) are combined to form a bigger item (q_i, q_j), in which at most $8n^3$ items are generated (at most two nodes each for i, k and j). See Supplementary Fig. 7c for the illustration of above analysis; see lines 22–25 in Supplementary Fig. 2 and lines 20–24 in Supplementary Fig. 4 for the pseudocode of the bifurcation case in our work.

By contrast, CDSfold defines each item as $(i, j, \mathrm{nuc}_i, \mathrm{nuc}_j)$, where nuc_i and nuc_j are the nucleotides at positions i and j, respectively. The number of items stored in CDSfold scales $16n^2$, because there are at most 4 nucleotide types for each nuc_i and nuc_j. For the bifurcation rule $S \rightarrow S\,P$, items $(i, k, \mathrm{nuc}_i, \mathrm{nuc}_k)$, and $(k+1, j, \mathrm{nuc}_{k+1}, \mathrm{nuc}_j)$, are combined to form $(i, j, \mathrm{nuc}_i, \mathrm{nuc}_j)$, in which at most $128n^3$ items are generated (at most 4×4 nucleotide types at nuc_i and nuc_j, and 4×2 nucleotide pairs between nuc_k and nuc_{k+1}). See Supplementary Fig. 7d for the analysis illustration of CDSfold.

Compared to CDSfold, our work largely reduces the time complexity constant of the bifurcation rule $S \rightarrow S\,P$ from 128 to 8. The cubic-time bifurcation rule which dominates the runtime in CDSfold is greatly accelerated in our algorithm. Empirically, our algorithm scales quadratically rather than cubically with mRNA sequence length for practical applications (Fig. 3 and Supplementary Fig. 8).

Joint optimization of stability and codon optimality. Codon optimality is an important factor in mRNA design, which should be jointly optimized with stability[5], and our work is the first to solve this joint optimization problem, thanks to the DFA representation and lattice parsing. By contrast, previous work (Cohen et al. and CDSfold) does not perform, and is impossible to be extended to perform, such a joint optimization. First, Cohen et al. only optimize stability without considering codon optimality. CDSfold uses simulated annealing to improve CAI by fine-tuning from the MFE solution, but this is a heuristic with no guarantees. Second, CDSfold's objective function, $\mathrm{MFE} \cdot \mathrm{CAI}^{\lambda}$, is impossible for dynamic programming due to the difference between MFE and CAI, where MFE is a sum of free energy for each component substructure (additive) but CAI is a geometric mean of the relative codon usages (multiplicative). To reconcile this difference, our formulation defines a novel objective that factors the logarithm of CAI for an mRNA additively onto its individual codons, thus making it decomposable and amenable to dynamic programming (see 'The LinearDesign algorithm' for details). By contrast, CDSfold's objective formulation does not factor into individual codons, and thus cannot be incorporated into global optimization. Last but most importantly, even if CDSfold were to borrow our formulation, its fundamental codon representation still rules out joint optimization. Our framework easily encodes (the logarithm of) CAI in our DFA representation, for example, we can integrate CAI onto a weighted DFA for leucine (Supplementary Fig. 6a). By contrast, CDSfold has to use an extended nucleotides representation for codon choices, which makes it impossible to do joint optimization with CAI (Supplementary Fig. 6b,c).

To summarize, our framework easily incorporates codon optimality into the joint optimization that previous work did not (and could not be extended to) tackle. Our objective integrates (the logarithm of) CAI and MFE together, while the objective of CDSfold is not able to reconcile these two factors. Furthermore, even if using our objective formulation, CDSfold's representation of codon choices still rules out the possibility of CAI integration.

Generalizability. Our DFA framework is so general that it can also represent arbitrary (non-standard) genetic codes, modified nucleotides, and coding constraints such as adjacent codon pair preference, which previous work could not handle even with major modifications. See 'DFAs for other genetic codes, coding constraints and modified nucleotides' for details.

Linear-time version for long sequence and suboptimal candidates. We further develop a faster, linear-time, approximate version which greatly reduces runtime for long sequences with small sacrifices in search quality, which we also use to generate multiple suboptimal candidates with varying folding stability and codon optimality as candidates for experimentation.

Verification of wet laboratory experiments. Extensive experiments confirm that compared to the standard codon-optimization benchmark, our designs are substantially better in chemical stability and protein expression in vitro, and the corresponding mRNA vaccines elicit up to 128 times higher antibody responses in vivo.

Another recent work[29] optimized mRNA designs and screened them via an experimental platform. LinearDesign had a central role in their work as the starting point of their optimizations (see figure 4b of their paper), followed by fine-tunings by both human players and a Monte Carlo tree search algorithm. The resulting

coding regions are flanked by different UTRs, and then tested on stability and protein expression. LinearDesign-generated sequences showed strong stability and protein expression results with different UTRs (figures 2g and 4a of their paper), independently confirming our in vitro experiments. However, they did not perform any in vivo validations.

Benchmark dataset and machine. To estimate the time complexity of LinearDesign, we collected 114 human protein sequences from UniProt[24], with lengths from 78 to 3,333 amino acids (not including the stop codon); see Supplementary Table 1. We benchmarked LinearDesign on a Linux machine with 2 Intel Xeon E5-2660 v3 CPUs (2.60 GHz) and 377 GB memory, and used Clang (11.0.0) to compile. The code only uses a single thread.

Additional design constraints. Some studies have shown that protein expression level drops if the 5′-end leader region has more secondary structure[5,50–53]. To design sequences with less structures at 5′-end leader region, we take a simple 'design, enumerate and concatenate' strategy to avoid structure in the leader region: (1) design the CDS region except for the 5′-end leader region (that is, the first 15 nucleotides); (2) enumerate all possible subsequences in the 5′-end leader region; and (3) concatenate each subsequence with the designed sequence, refold, and choose the one whose 5′-end leader region has the most unpaired nucleotides.

In addition, it has been revealed that long double-stranded regions may induce unwanted innate immune responses by previous studies[27,54,55]. Considering this, we do not allow long double-stranded regions that include 33 or more base pairs by adding this constraint in the design process.

RNA secondary structure prediction and visualization. Vienna RNAfold from ViennaRNA package (version 2.4.14) is used for predicting and drawing the secondary structure of mRNA sequence, and calculating the MFE of secondary structures.

In vitro and in vivo experiments

Preparation of mRNA and its formulation. mRNA molecules were synthesized in vitro by T7 RNA polymerase using linearized plasmid as DNA template. The open reading frame region is flanked with the 5′ and 3′ UTRs followed by a 70-nt poly-A tail. For all spike protein-coding sequences, the in vitro transcription reaction was conducted at 37 °C for 4 h, followed by digestion with DNase I (Hongene Biotech). mRNA encoding full-length spike protein without proline substitution was then capped using Vaccinia Capping Enzyme (Hongene Biotech) and purified with magnetic Dynabeads (Thermo Fisher). Eluted mRNA was further treated with Antarctic Phosphatase (Hongene Biotech) at 37 °C for 30 min to remove residual 5′-triphosphates. For all VZV gE sequences, mRNA was co-transcriptionally capped using m7(3′OMeG)(5′)ppp(5′)(2′OMeA)pG capping reagent (Hongene Biotech) in a 'one-pot' reaction at 30 °C for 16 h, followed by treatment with DNase I. Capped mRNA encoding spike or VZV gE protein was then purified using beads. For the preparation of formulated mRNA vaccines, lipopolyplex (LPP) formulation was used to encapsulate mRNA cargo as described previously[56]. LPP is a lipid-based mRNA delivery system and has been demonstrated to provide high efficacy and good safety profile[26].

Agarose gel electrophoresis and integrity assay of mRNA. To study the electrophoretic mobility profile of mRNA molecules, mRNA samples suspended in Ambion RNA storage buffer (Thermo Fisher) were denatured at 75 °C for 5 min and snap-cooled on ice before being loaded onto 1% non-denaturing agarose gel (130 V for 1 h at room temperature). Gel image was taken by Gel Doc XR+ Gel Documentation System (Bio-Rad).

To assess the in-solution stability of mRNA, samples were incubated in PBS buffer containing 10 mM Mg^{2+}. Sampling was conducted at time points (0, 1, 2, 4, 8, 12, 16, 24, 32, 48 and 60 h). For a faster degradation process, PBS buffer containing 20 mM Mg^{2+} instead of 10 mM was used. Sampling was done in a relatively shorter time span (0, 1, 2, 4, 8, 12, 15, 18, 21, and 24 h). RNA integrity was analysed by Qsep100 Capillary Electrophoresis System. The integrity was represented as the proportion of full-length mRNA calculated on electropherogram. The data were normalized to time point 0 h. To extrapolate the half-life of each sequence, one-phase decay equation:

$$Y = (Y_0 - \text{plateau}) \cdot e^{-KX} + \text{plateau}$$

was used to fit the curve. The Y_0 and plateau were set as 100 and 0, respectively. Half-life was computed as $\ln(2)/K$, where K refers to decay rate constant.

Protein expression assay. HEK293 cells (ATCC) were cultured in Dulbecco's Modified Eagle's Medium (DMEM) (Hyclone) containing 10% fetal bovine serum (FBS) (GEMINI) and 1% penicillin-streptomycin (Gibco). All cells were cultured at 37 °C in a 5% CO_2 condition.

For the measurement of protein expression, cells were transfected with mRNA using Lipofectamine MessengerMAX (Thermo Scientific). In brief, a mix of 2 μg mRNA and 6 μl of Lipofectamine reagent was prepared following the manual instructions and then incubated with cells for 24 or 48 h. For flow cytometric analysis, cells were collected and stained with Live/Dead cell dye (Fixable Viability Stain 510, BD) for 5 min. After washing, cells were incubated with anti-spike receptor-binding domain (RBD) chimeric monoclonal antibody (1:100 dilution, Sino Biological) for 30 min, followed by washing and incubation with PE-anti-human IgG Fc (1:100 dilution, Biolegend) for 30 min. Samples were analysed on BD Canto II (BD Biosciences). Data were processed using FlowJo V10.1 (Tree Star).

In vivo immunogenicity study. C57BL/6 mice (6 to 8 weeks of age) were intramuscularly immunized twice with 10 μg LPP-formulated mRNA vaccines at a 2-week interval. Sera and spleens were collected 14 days after boost shot.

Surrogate virus neutralization assay. Neutralizing antibody titre was measured using surrogate virus neutralization assay as previously described[57], with some modifications. In brief, 96-well plates (Greiner Bio-one) were coated with recombinant with human ACE2 protein (100 ng per well, Genscript) overnight at 4 °C. Plates were washed with 1 × PBS-T and blocked with 2% BSA for 2 h at room temperature. HRP-conjugated RBD (100 ng ml^{-1}) was incubated with serially diluted serum from immunized mice at an equal volume (60 μl each) for 30 min at 37 °C. Sera collected from PBS-treated mice were used as negative control. Then a 100 μl mixture of RBD and serum was added into each well and incubated for 15 min at 37 °C. After washing, TMB substrate (Invitrogen) was used for colour development and the absorbance at 450 nm was recorded using BioTek microplate reader. The IC_{50} value was calculated using four-parameter logistic non-linear regression.

Enzyme-linked immunosorbent assays. In brief, recombinant SARS-CoV-2 spike ectodomain protein or VZV gE protein (Genscript) diluted in coating buffer (Biolegend) were used to coat 96-well EIA/RIA plates (Greiner Bio-one, 100 ng per well) at 4 °C overnight. The plates were then washed with 1×PBS-T (0.05% Tween-20) and blocked with 2% BSA in PBS-T for 2 h at room temperature. Serum samples with serial dilutions were added and incubated for 2 h at room temperature. After washing, HRP-conjugated goat anti-mouse IgG antibody (1:10,000) was added and incubated for 1 h. TMB substrate (Invitrogen) was then used for colour development and the absorbance was read at 450 nm using BioTek microplate reader. End-point titres were calculated as

the largest sample dilution factor yielding a signal that exceeds 2.1-fold value of the background[58].

ELISpot assay. Frequency of spike (or VZV gE) antigen-specific IFNγ-secreting T cells was evaluated using Mouse IFNγ ELISpotplus Kit (Mabtech) according to the manual. In brief, 3×10^5 mouse splenocytes were added to wells pre-coated with anti-mouse IFNγ capturing antibodies and were incubated with spike protein or VZV gE peptide pool (10 μg ml^{-1}) for 20 h. After washing, plates were incubated with Streptavidin–alkaline phosphatase (1:1,000) for 1 h at room temperature. Spots were developed with BCIP/NBT substrate solution and counted using Immunospot S6 analyzer (CTL). Due to multiple steps and exponential change of antibody and antigen-specific T cells during the immunity induction process, in vivo immunogenicity data usually have high data variations. Inoculation of mRNA vaccine involves extra processes such as tissue transfection and protein translation, and the variations in these process efficiencies together with variable dosing and differences in individual mouse's immune status usually bring more immunogenicity variations than protein-based vaccines. From our experience, the variations observed in this study are typical for mRNA vaccines.

Ethics statement. All mouse studies were performed in strict accordance with the guidelines set by the Chinese Regulations of Laboratory Animals and Laboratory Animal-Requirements of Environment and Housing Facilities. Animal experiments were carried out with the approval from the Institutional Animal Care and Use Committee (IACUC) of Shanghai Model Organisms Center.

Statistics and reproducibility. Geometric means or arithmetic means are represented by the heights of bars, or symbols, and error bars represent the corresponding s.d. Two-tailed Mann–Whitney U tests were used to compare two experimental groups for the in vivo studies. To compare more than two experimental groups, one-way ANOVA with Dunn's multiple comparisons tests were applied in the in vitro protein expression experiment. Statistical analyses were performed using Prism v.8 (GraphPad). $*P < 0.05$, $**P < 0.01$, $***P < 0.001$. The raw P values from the statistical analysis are summarized in the figshare file (https://doi.org/10.6084/m9.figshare.22193251). In vitro experiments were independently repeated in triplicate. Animal experiments were completed once. Gel electrophoresis experiment was repeated three times to obtain similar results.

Data reporting. No statistical methods were used to predetermine sample size. The experiments were not randomized. The investigators were not blinded to allocation during experiments and outcome assessment.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The UniProt sequences used to estimate the time complexity of LinearDesign are included in Supplementary Table 1 and are deposited at our figshare repository (https://doi.org/10.6084/m9.figshare.22193251). The SARS-CoV-2 spike and VZV gE protein-coding sequences and UTR sequences used in the biological experiments are included at the end of the Supplementary Information file and are available at our figshare repository. Source data are provided with this paper.

Code availability

The LinearDesign source code is available to all parties on GitHub (https://github.com/LinearDesignSoftware/LinearDesign) and Zenodo (https://doi.org/10.5281/zenodo.7839739), and is free for academic and research use.

39. Rivas, E. The four ingredients of single-sequence RNA secondary structure prediction. a unifying perspective. *RNA Biol.* **10**, 1185–1196 (2013).
40. Nussinov, R. & Jacobson, A. B. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proc. Natl Acad. Sci. USA* **77**, 6309–6313 (1980).
41. Kasami, T. *An Efficient Recognition and Syntax-Analysis Algorithm for Context-Free Languages*, Coordinated Science Laboratory Report no. R-257 (Univ. Illinois–Urbana, 1966).
42. Younger, D. H. Recognition and parsing of context-free languages in time n^3. *Inf. Control* **10**, 189–208 (1967).
43. Rivas, E., Lang, R. & Eddy, R. A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more. *RNA* **18**, 193–212 (2012).
44. Huang, L., Fayong, S. & Guo, Y. Structured perceptron with inexact search. *In Proc. 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 142–151 (Association for Computational Linguistics, 2012).
45. Lorenz, R. et al. ViennaRNA package 2.0. *Algorithms Mol. Biol.* **6**, 26 (2011).
46. Turner, D. H. & Mathews, D. H. NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.* **38**, D280–D282 (2010).
47. Kawaguchi, Y., Honda, H., Taniguchi-Morimura, J. & Iwasaki, S. The codon CUG is read as serine in an asporogenic yeast *Candida cylindracea*. *Nature* **341**, 164–166 (1989).
48. Bonitz, S. G. et al. Codon recognition rules in yeast mitochondria. *Proc. Natl Acad. Sci. USA* **77**, 3167–3170 (1980).
49. Gamble, C. E., Brule, C. E., Dean, K. M., Fields, S. & Grayhack, E. J. Adjacent codons act in concert to modulate translation efficiency in yeast. *Cell* **166**, 679–690 (2016).
50. Ding, Y. et al. In vivo genome-wide profiling of RNA secondary structure reveals novel regulatory features. *Nature* **505**, 696–700 (2014).
51. Wan, Y. et al. Landscape and variation of RNA secondary structure across the human transcriptome. *Nature* **505**, 706–709 (2014).
52. Shah, P., Ding, Y., Niemczyk, M., Kudla, G. & Plotkin, J. B. Rate-limiting steps in yeast protein translation. *Cell* **153**, 1589–601 (2013).
53. Tuller, T. & Zur, H. Multiple roles of the coding sequence 5′ end in gene expression regulation. *Nucleic Acids Res.* **43**, 13–28 (2014).
54. Husain, B., Mukerji, I. & Cole, J. L. Analysis of high-affinity binding of protein kinase R to double-stranded RNA. *Biochemistry* **51**, 8764–8770 (2012).
55. Hur, S. Double-stranded RNA sensors and modulators in innate immunity. *Annu. Rev. Immunol.* **37**, 349–375 (2019).
56. Persano, S. et al. Lipopolyplex potentiates anti-tumor immunity of mRNA-based vaccination. *Biomaterials* **125**, 81–89 (2017).
57. Tan, C. W. et al. A SARS-CoV-2 surrogate virus neutralization test based on antibody-mediated blockage of ACE2–spike protein–protein interaction. *Nat. Biotechnol.* **38**, 1073–1078 (2020).
58. McKay, P. F. et al. Self-amplifying RNA SARS-CoV-2 lipid nanoparticle vaccine candidate induces high neutralizing antibody titers in mice. *Nat. Commun.* **11**, 3523 (2020).

Acknowledgements The authors thank R. Das (Stanford) for introducing the mRNA design problem to us; R. Li (Baidu) for connecting Baidu Research with StemiRNA; J. Li (Baidu Research) for coordinating resources for this project; G. Terai and K. Asai (Univ. of Tokyo) for sending us the CDSfold code; S. Aviran (UC Davis) for spotting a typo in the hyper-parameter λ in our earlier version; A. Solórzano (Pfizer) for the question on LinearDesign's independence of the choice of UTRs; J. Lin (Fudan) for early discussions; and S. Li (Oregon State Univ.) for proofreading and help with Latex. We acknowledge the assistance from L. Huang and M. Liu (StemiRNA) in LPP formulation of mRNA vaccines and help from other colleagues, including Y. Yi, Q. Wang, W. Wang and Y. Ge with in vivo studies. We thank Sanofi and many other vaccine companies worldwide for licensing and early adoption of LinearDesign. D.H.M. is supported by National Institutes of Health grant R35GM145283. A.L. was an employee of StemiRNA and is currently supported by the Natural Science Foundation of Jiangsu Province (BK20221031), the National Science Foundation of China (32200764, 82061138008) and the Fundamental Research Funds for the Central Universities (2632022YC01). C.X. was sponsored by Shanghai Pujiang Talent Program (22PJ1423100). The funding from StemiRNA Therapeutics was supported by the Science and Technology Commission of Shanghai Municipality, China (20S11909100, 22S11902300); Shanghai Strategic Emerging Industry Development special fund (ZJ640070216) and the project of mRNA Innovation and Translation Center, Shanghai, China.

Author contributions L.H. conceived and directed the project. L.H. designed the basic algorithm for the Nussinov model and wrote a Python prototype, and H.Z. and L.Z. extended this algorithm to the Turner model and implemented it in C++, which Z.L. optimized. L.H., H.Z. and L.Z. designed the CAI integration algorithm, which L.Z. and H.Z. implemented. L.Z. implemented the beam search and handled design constraints. K.L. made the web server. B.L. implemented a baseline. Y.Z. and H.L. supervised the in vitro and in vivo experiments. C.C. performed the mRNA synthesis and gel electrophoresis experiments. A.L., C.X., H.J., X.M. and F.Z. performed the protein expression and in vivo assays, and C.X. performed chemical stability and structure compactness assays. D.H.M. discussed the approach and provided guidance for in silico analysis and writing. L.H., H.Z., L.Z., D.H.M., A.L., C.X., H.S., H.L. and Y.Z. wrote the manuscript.

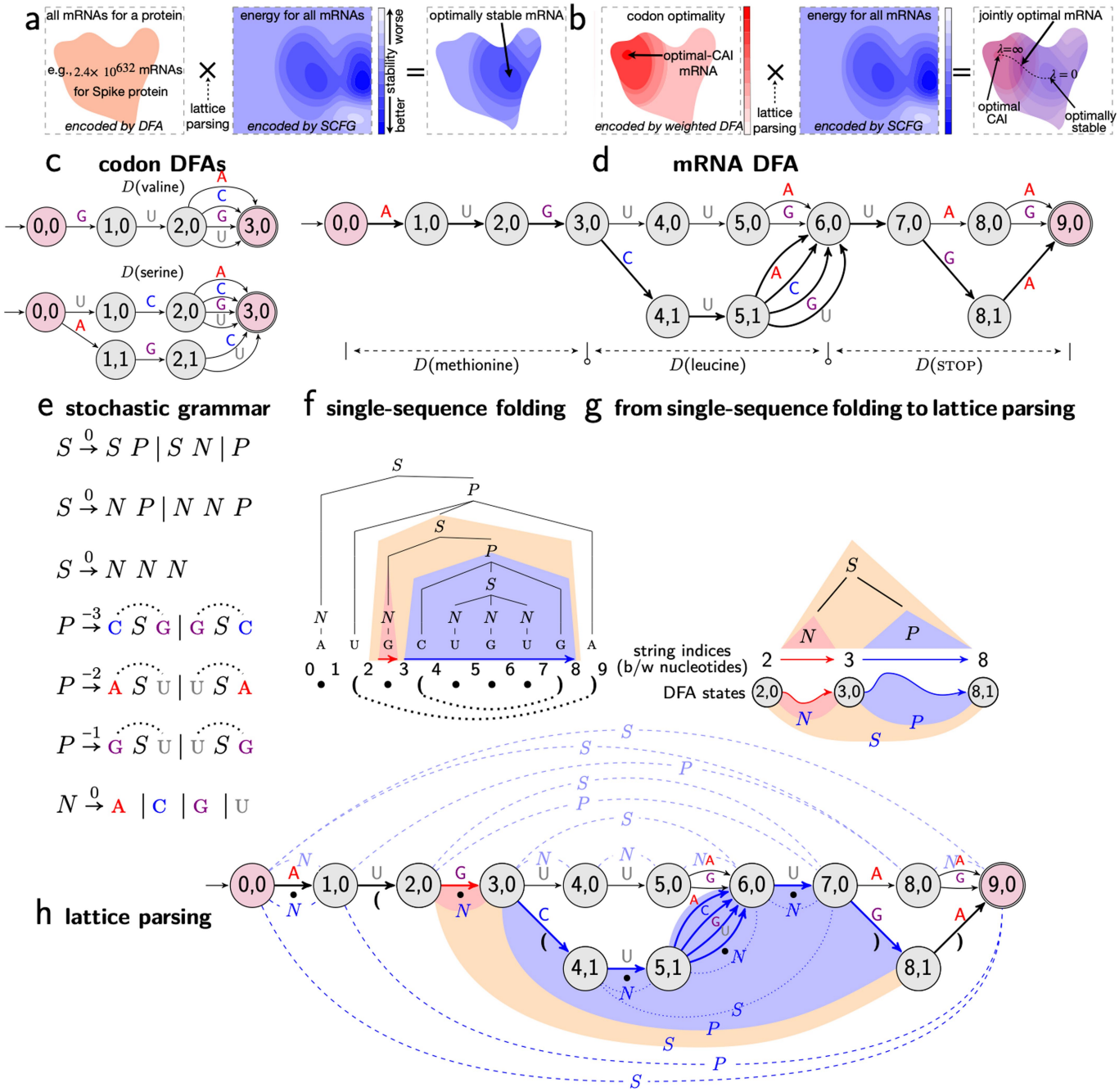
Competing interests Baidu USA filed a patent for the LinearDesign algorithm in 2021 listing H.Z., L.Z., Z.L., K.L., B.L. and L.H. as inventors. The work of H.Z., L.Z., Z.L., K.L., B.L. and L.H. for the development of the LinearDesign algorithm was conducted at Baidu USA. StemiRNA Therapeutics has filed a provisional patent for the VZV mRNA vaccine listing C.X., H.S. and H.L. as inventors. Sanofi entered a non-exclusive licensing agreement with Baidu USA in 2021 to use LinearDesign to develop mRNA vaccines and therapeutics. H.Z., L.Z., Z.L., K.L., B.L. and L.H. are or were employees of Baidu USA. A.L., C.X., X.M., F.Z., H.J., C.C., H.S., H.L. and Y.Z. are or were employees of StemiRNA Therapeutics. L.H. and D.H.M. are also cofounders of Coderna.ai.

Additional information

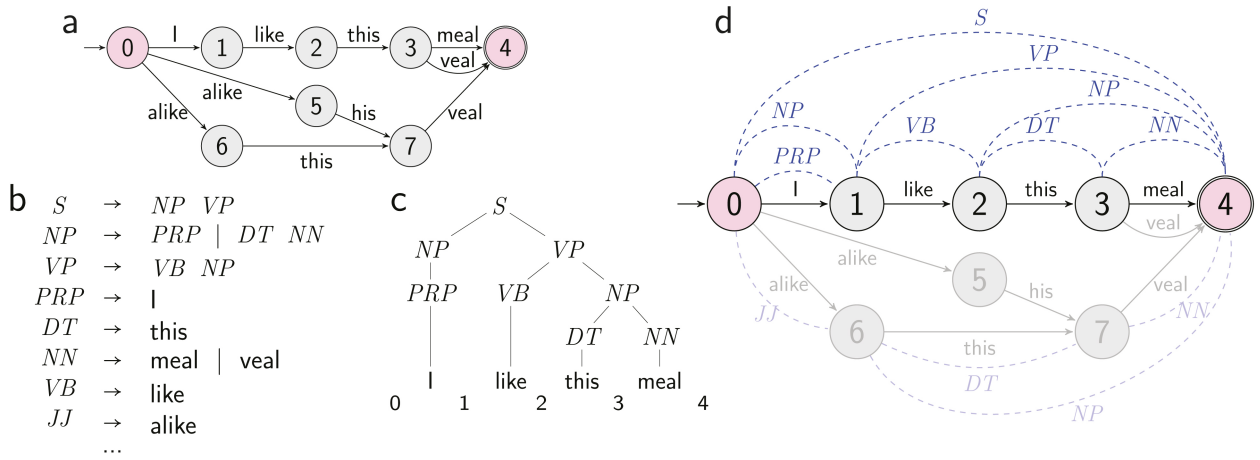
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41586-023-06127-z.

Correspondence and requests for materials should be addressed to Liang Huang, Yujian Zhang, David H. Mathews or Hangwen Li.

Peer review information *Nature* thanks Anna Blakney and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.



Extended Data Fig. 1 | Illustrations of the optimization problems in mRNA design, DFA representations, single sequence folding as natural language parsing, and lattice parsing. a–**b**, Visualization of mRNA design as optimization problems for stability (objective 1, in **a**) and joint stability and codon optimality (objectives 1 and 2, in **b**). **c**–**h** show how lattice parsing solves the first optimization problem (see Fig. 2d for the second). **c**, Codon DFAs. **d**, An mRNA DFA made of three codon DFAs. The thick paths depict the optimal mRNA sequences under the simplified energy model in **e**, AUGCU⋆UGA, where ⋆ could be any nucleotide. **e**, Stochastic context-free grammar (SCFG) for a simplified folding free energy model. Each rule has a cost (i.e., energy term, the lower the better), and the dotted arcs represent base pairs in RNA secondary structure. **f**, Single-sequence folding is equivalent to context-free parsing with an SCFG; the parse tree represents the best secondary structure for the input mRNA sequence. **g**, We extend single-sequence parsing (top) to lattice parsing (bottom) by replacing the input string with a DFA, where each string index becomes a DFA state, and a span becomes a path between two states. **h**, Lattice parsing with the grammar in **e** for the DFA in **d**. The blue arcs below the DFA depict the (shared) best structure for the optimal sequences AUGCU⋆UGA in the whole DFA, while the dashed light-blue arcs above the DFA represent the best structure for a suboptimal sequence AUGUUAUAA. Lattice parsing can also incorporate codon optimality (objective 2, see **b**), by replacing the DFA with a weighted one (Fig. 2d).



e

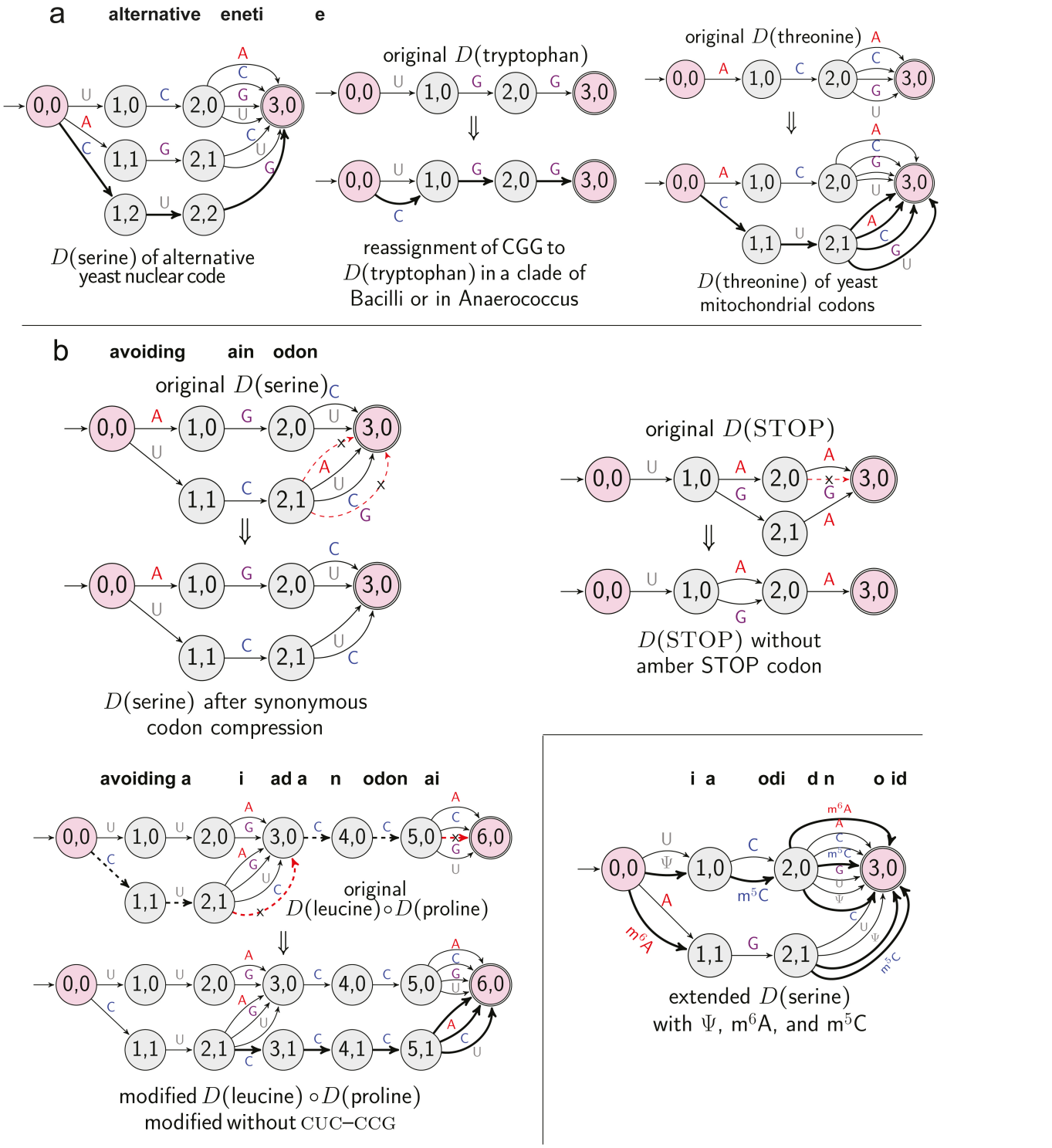
well-known connections in the literature:

single-sequence parsing	⇔	single-sequence RNA folding
CKY ~1965, $O(n^3)$	⇔	Nussinov/Zuker ~1980, $O(n^3)$

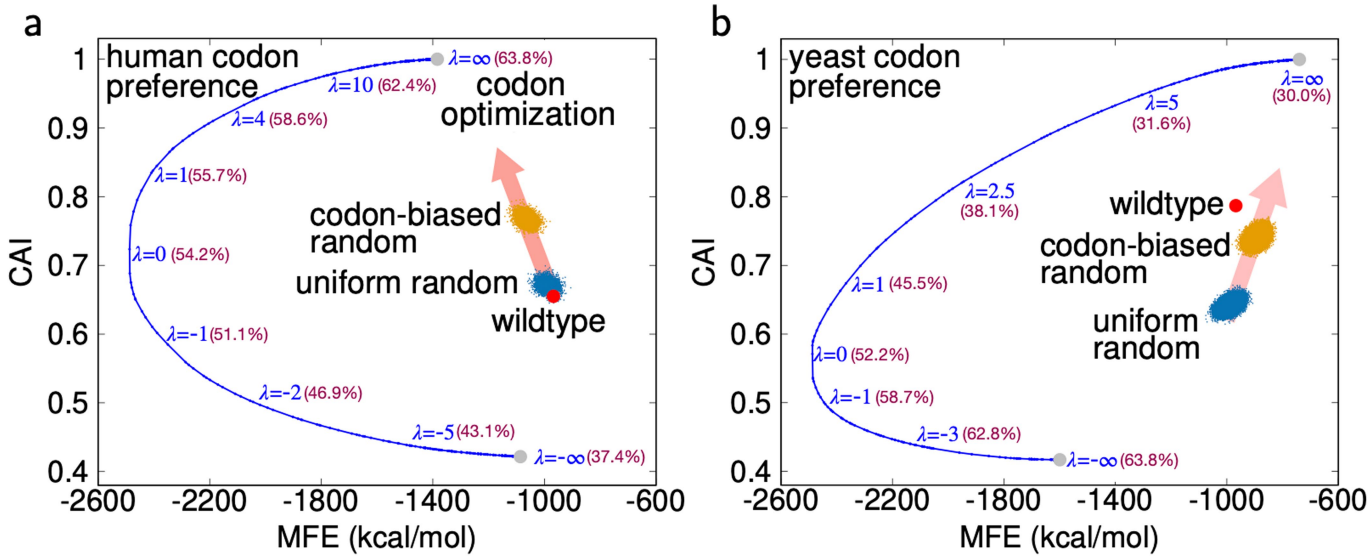
new connections in this work:

lattice parsing	⇔	mRNA design
Bar-Hillel *et al*, 1961, $O(\|D\|^3)$	⇔	our work, $O(n^3)$

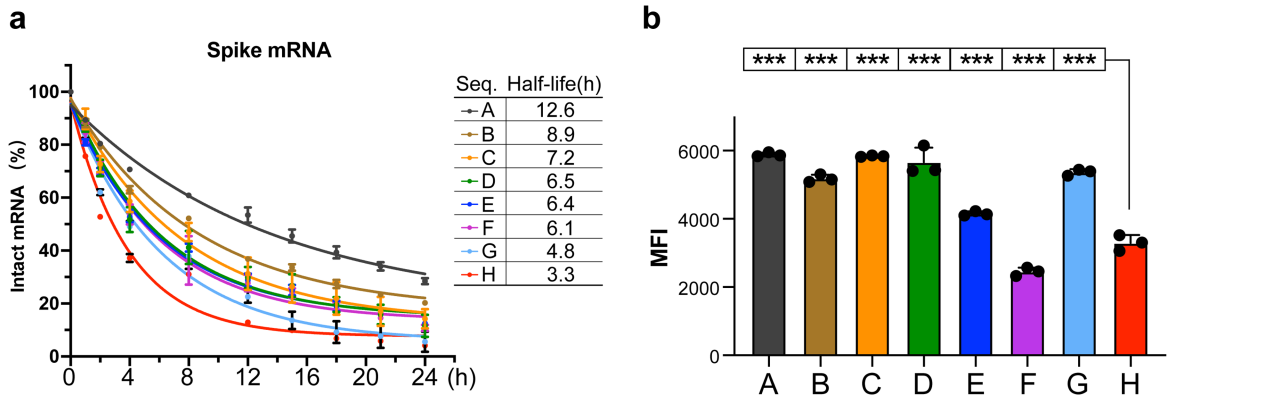
Extended Data Fig. 2 | Word lattice and lattice parsing in natural language processing, and correspondence between linguistics and biology. a, An example of word lattice (sentence DFA) for speech recognition. **b**, Simplified language grammar. **c**, Single sentence parsing with between-word indices, which is a special case of word lattice parsing. **d**, Illustration of word lattice parsing for speech recognition with given word lattice and language grammar; the dashed blue arcs above the DFA depict the best parsing structure for the optimal sentence "I like this meal", while the dashed light-blue arcs below the DFA represent the best parsing structure for a non-optimal sentence "alike this veal". **e**, Correspondence between computational linguistics (left) and computational biology (right). See also Fig. 1.



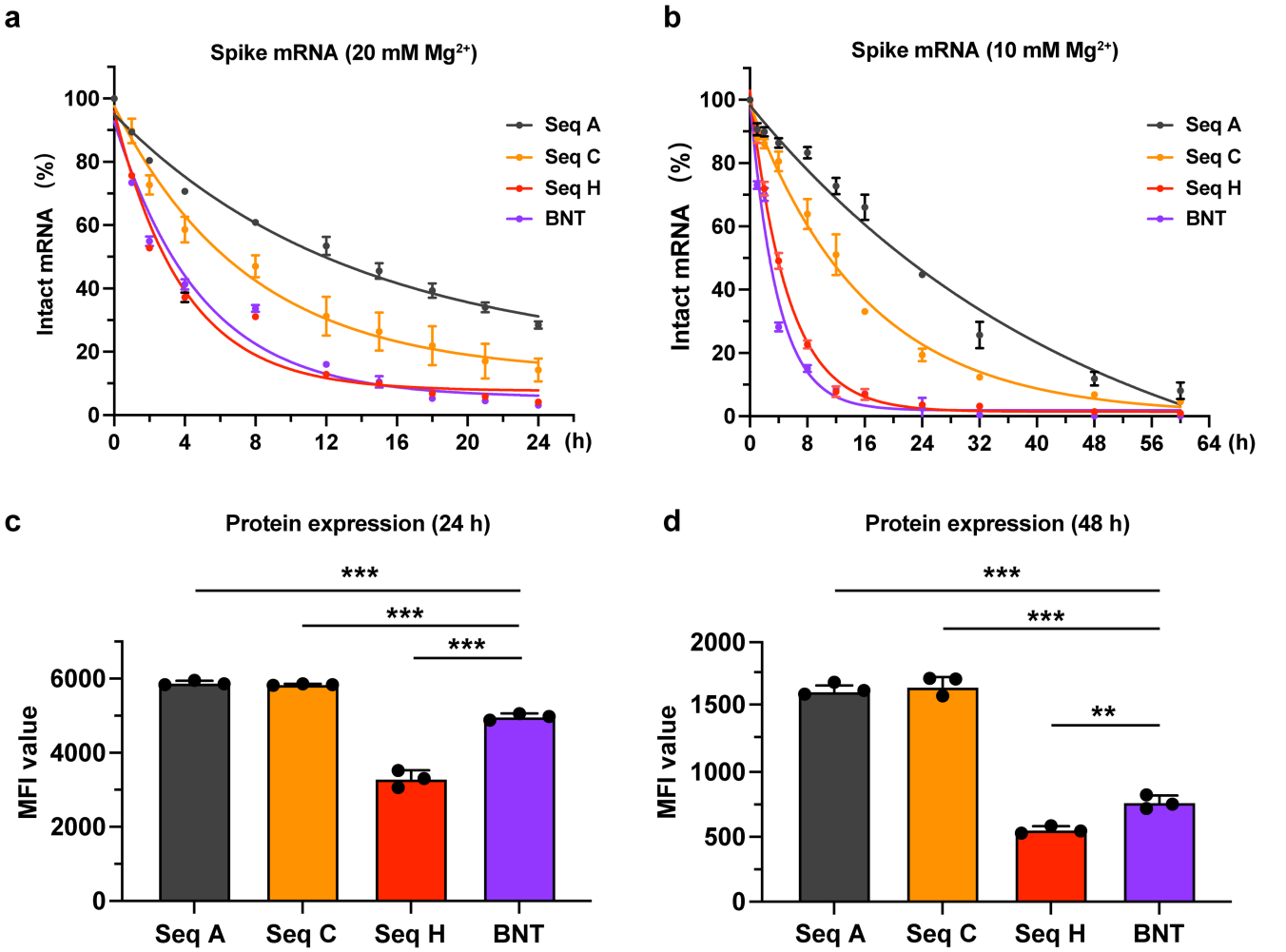
Extended Data Fig. 3 | Examples of the DFA representations for extended codons, modified nucleotides, and coding constraints. a, Alternative genetic codes of serine, tryptophan, and threonine. **b**, Avoiding certain codon. On the left it shows the original DFA of serine (up), in which the red dashed arrows indicating UCA and UCG are chosen to be avoided, resulting in a new DFA (down). On the right it shows removing the rare amber STOP codon (UAG). **c**, Avoiding a specific adjacent codon pair. **d**, Extended serine DFA can include chemically modified nucleotides pseudouridine (Ψ), 6-Methyladenosine (m^{6}A) and 5-methylcytosine (m^{5}C).



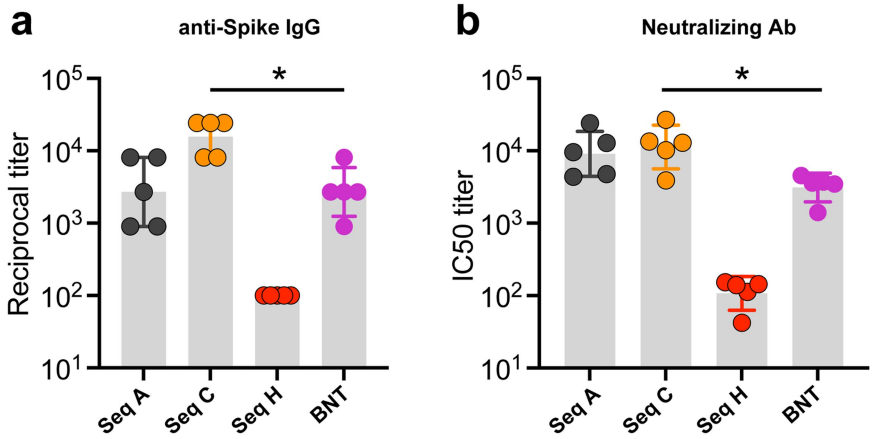
Extended Data Fig. 4 | Two dimensional (MFE-CAI) visualizations of mRNA designs for the Spike protein using human codon preference (a) and yeast codon preference (b) with positive and negative λ's. GC% are shown in parentheses. The human genome prefers GC-rich codons that lead to higher CAI designs are with higher GC%, while the yeast genome prefers AU-rich codons that exhibit an opposite relationship between CAI and GC%. See also Fig. 3 for more *in silico* results of LinearDesign.



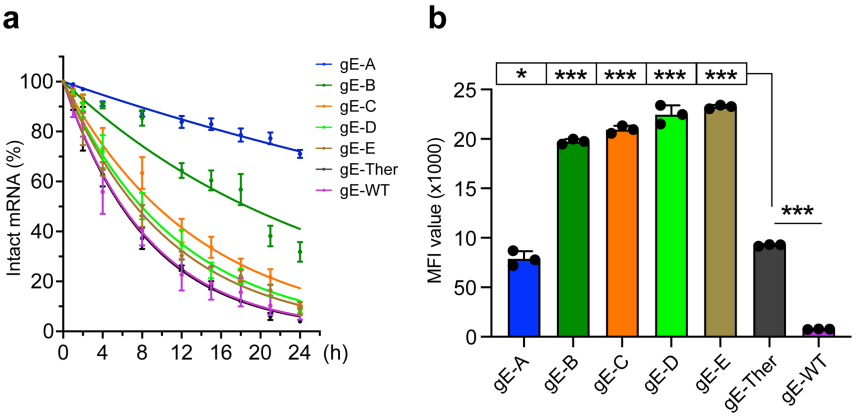
Extended Data Fig. 5 | Extra experimental results of LinearDesign-generated mRNAs encoding the Spike protein. a, In-solution stability of sequences A–H in PBS buffer containing 20 mM Mg^{2+} at 37 °C over the course of 24 h. The degradation experiments were performed in triplicate independently and the data were presented as mean ± s.d. and fitted with a one-phase decay curve. **b**, Protein expression of mRNAs following transfection into HEK293 cells for 24 h was determined by flow cytometry. MFI values derived from three independent experiments are shown. Kruskal–Wallis analysis of variance (ANOVA) with Dunn's multiple comparisons test to H group was performed for statistical analysis. ***$P < 0.001$. See Fig. 4c, d for similar experiments but with 10 mM Mg^{2+} and 48 h, respectively.



Extended Data Fig. 6 | In-solution stability and protein expression of sequences A, C, H and BNT, for a head-to-head in vitro comparison between LinearDesign and BioNTech-Pfizer mRNA sequences. a-b, In-solution stability of mRNAs in PBS buffer containing 20 mM Mg^{2+} or 10 mM Mg^{2+} at 37 °C. Data are from three independent experiments and were presented as mean ± s.d. and fitted with one-phase decay curve. **c-d**, Protein expression of mRNAs was determined 24 h or 48 h following transfection into HEK293 cells. MFI value is presented as mean ± s.d. Each group has three independent assays and 10,000 live cells were collected for analysis in each assay. Kruskal–Wallis analysis of variance (ANOVA) with Dunn's multiple comparisons test to BNT group was performed for statistical analysis. $^{**}P < 0.01$, $^{***}P < 0.001$.



Extended Data Fig. 7 | Antibody (Ab) responses induced by sequences A, C, H and BNT-based mRNA vaccines, for a head-to-head in vivo comparison between LinearDesign and BioNTech-Pfizer mRNA sequences. C57BL/6 mice ($n = 5$) were immunized *i.m.* with two doses of mRNA vaccines at a 2-week interval. Seven days after boost immunization, levels of anti-Spike IgG (**a**) and neutralizing Abs (**b**) against pseudotyped SARS-CoV-2 were measured. Data were presented as geometric mean ± geometric s.d. A two-tailed Mann-Whitney U test was used for statistical analysis. *$P < 0.05$. See Source Data for details.



Extended Data Fig. 8 | Extra stability and protein expression results of LinearDesign-generated mRNAs encoding VZV gE protein. a, In-solution stability of mRNAs upon incubation in buffer (Mg^{2+} = 20 mM) at 37 °C. Percentage of intact mRNA is shown. Data are presented as mean ± SD from three independent experiments. **b**, Protein expression of mRNAs following transfection into HEK293 cells for 24 h was determined by flow cytometry. Each group has three independent assays and 10,000 live cells were collected for analysis in each assay. MFI value is presented as mean ± s.d. Kruskal–Wallis analysis of variance (ANOVA) with Dunn's multiple comparisons test to gE-Ther group was performed for statistical analysis. *$P < 0.05$, ***$P < 0.001$.

Extended Data Table 1 | The LinearDesign-generated coding-region sequences, due to more secondary structures within the coding region, are less likely to form base pairs with or interfere with the structures of the UTRs

		UTRs																			
sequence of CDS	MFE of CDS	StemiRNA COVID-19				BioNTech				Moderna				CureVac				human β-globin			
		MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'
	kcal/mol					*kcal/mol*				*kcal/mol*				*kcal/mol*				*kcal/mol*			
A	-2,287.3	-2,328.9	11	9	2	-2378.2	9	5	4	-2,334.1	11	10	1	-2,314.6	1	0	1	-2,328.3	8	0	8
B	-2,213.2	-2,252.8	12	10	2	-2302.2	6	5	1	-2,258.9	13	12	1	-2,236.9	1	0	1	-2,251.1	29	0	29
C	-2,206.0	-2,243.5	11	9	2	-2294.3	7	6	1	-2,248.9	9	8	1	-2,230.6	1	0	1	-2,245.2	10	7	3
D	-1,967.4	-2,004.6	13	7	6	-2057.1	10	5	5	-2,011.5	11	10	1	-1,992.6	11	11	0	-2,005.7	8	0	8
E	-1,961.3	-2,003.1	13	11	2	-2057.5	16	5	11	-2,009.7	15	14	1	-1,989.9	4	3	1	-2,002.1	8	0	8
F	-1,969.3	-2,009.9	11	9	2	-2061.1	12	5	7	-2,012.8	11	10	1	-1,995.3	9	9	0	-2,009.1	19	0	19
G	-1,639.3	-1,680.4	34	7	27	-1742.1	62	4	58	-1,688.5	62	0	62	-1,674.1	25	25	0	-1,688.1	23	0	23
H	-1,244.4	-1,287.6	31	16	15	-1346.3	66	8	58	-1,292.9	18	17	1	-1,285.6	77	52	25	-1,286.4	21	0	21
CureVac	-1384.4	-1,423.0	14	8	6	-1478.5	64	5	59	-1,432.3	65	19	46	-1,419.1	77	61	16	-1,425.7	26	0	26
Moderna	-1,369.2	-1,411.1	41	7	34	-1464.3	66	6	60	-1,422.2	60	12	48	-1,406.3	53	45	8	-1,414.0	27	0	27
BioNTech	-1,217.2	-1,265.4	34	5	29	-1316.1	98	6	92	-1,269.2	46	15	31	-1,253.7	58	54	4	-1,266.8	23	0	23
MFE-opt.	-2,486.7	-2,523.7	1	1	0	-2574.8	10	5	5	-2,530.9	1	1	0	-2,512.8	3	3	0	-2,522.5	7	0	7
CAI-opt.	-1,384.1	-1,421.7	34	11	23	-1478.2	59	0	59	-1,430.2	35	7	28	-1,420.4	53	53	0	-1,426.8	33	5	28
Wildtype	-966.7	-1011.7	18	16	2	-1060.6	33	21	12	-1,018.5	26	25	1	-999.9	68	45	23	-1,016.3	75	9	66

Here we show the numbers of predicted base pairs between UTRs and the coding regions of SARS-CoV-2 Spike protein. We used 5 different UTRs: StemiRNA COVID-19 UTRs used in wet lab experiments, BNT-162b2 (BioNTech) UTRs, mRNA-1273 (Moderna) UTRs, CV2CoV (CureVac) UTRs, and a widely used human β-globin mRNA UTRs. We tested 14 different sequences of the coding region: sequences A–H for wet lab experiments, sequences from three main mRNA vaccine companies, MFE-opt. and CAI-opt. sequences (i.e., sequences with the lowest folding free energy and with CAI=1, respectively), and the wildtype sequence. Most of the LinearDesign-generated mRNA sequences (sequences A–F and MFE-opt.) form fewer base pairs with UTRs. The folding free energies and structures are predicted by Vienna RNAfold (-d0 mode); MFEs of CDS are calculated without stop codon.

Extended Data Table 2 | Similar to Extended Data Table 1, LinearDesign-generated coding-regions for the VZV gE protein form less base pairs with the UTRs

		UTRs																			
sequence	MFE	StemiRNA VZV				BioNTech				Moderna				CureVac				human β-globin			
of CDS	of CDS	MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'	MFE	tot.	5'	3'
	kcal/mol					*kcal/mol*				*kcal/mol*				*kcal/mol*				*kcal/mol*			
gE-A	-1,145.6	-1,198.2	14	12	2	-1,239.0	8	5	3	-1,192.8	12	10	2	-1,183.3	5	5	0	-1,185.1	9	0	9
gE-B	-1,082.9	-1,134.8	15	13	2	-1,177.1	5	5	0	-1,126.5	14	12	2	-1,116.5	5	5	0	-1,123.0	29	0	29
gE-C	-932.3	-987.1	10	8	2	-1,026.4	6	6	0	-988.8	10	8	2	-966.8	17	17	0	-970.0	11	7	4
gE-D	-845.4	-910.1	13	9	4	-945.6	10	5	5	-909.3	12	10	2	-885.1	3	0	3	-892.3	14	0	14
gE-E	-805.0	-865.8	14	12	2	-907.8	15	5	10	-871.4	16	14	2	-843.8	19	12	7	-852.0	18	0	18
gE-Ther	-592.2	-662.2	11	9	2	-695.6	11	5	6	-649.8	6	0	6	-643.7	11	0	11	-641.5	9	0	9
gE-WT	-485.7	-546.9	23	5	18	-599.4	32	4	29	-544.9	61	0	61	-534.6	22	22	0	-529.8	46	11	35

Here we used 5 different UTRs: StemiRNA VZV UTRs used in wet lab experiments, BNT-162b2 (BioNTech) UTRs, mRNA-1273 (Moderna) UTRs, CV2CoV (CureVac) UTRs, and human β-globin mRNA UTRs. The 7 coding sequences are gE A–E, gE-Ther and gE-WT, which are used in the wet lab experiments of VZV. The folding free energies and structures are predicted by Vienna RNAfold (-d0 mode); MFEs of CDS are calculated without stop codon.

nature portfolio

Corresponding author(s): Liang Huang; Yujian Zhang; David H. Mathews; Hangwen Li

Last updated by author(s): Apr 6, 2023

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	The LinearDesign source code is available to all parties on GitHub (https://github.com/LinearDesignSoftware/LinearDesign), and is free for academic and research use.
Data analysis	Clang (11.0.0) is used to compile LinearDesign source code. Vienna RNAfold from ViennaRNA package (version 2.4.14; open source) is used for predicting and drawing the secondary structure of mRNA sequence, and calculating the Minimum Free Energy (MFE) of secondary structures. For the wet lab experiments, GraphPad Prism 8.0 was used for the data analysis. Flow cytometry data were analyzed by FlowJo 10.1.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

The UniProt sequences used to estimate the time complexity of LinearDesign are included in Supplementary Tab. 1 and deposited at our figshare repository https://

doi.org/10.6084/m9.figshare.22193251. The COVID-19 and VZV mRNA coding region sequences and UTR sequences used in the biological experiments are included at the end of Supplementary Information file and available on our figshare repository. Source data of the animal experiments is provided with this paper, and all source data of wet lab experiments is available on that repository.

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	N/A
Population characteristics	N/A
Recruitment	N/A
Ethics oversight	N/A

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	In the animal study, six mice (for COVID mRNA vaccine experiments) and five mice (for VZV mRNA vaccine experiments) were used in the corresponding experiments, respectively. The sample size of mice in each group was determined based on general animal study practice. Five or six mice per group were commonly used, which can also been seen in other publications (Nature 58, 567-571 (2020); Nat Commun 12, 2893 (2021); Molecular Therapy 29.6 (2021): 1970-1983.)
Data exclusions	There is no data exclusion in our study.
Replication	In vitro experiments were independently repeated in triplicate. All replication attempts were successful. Animal experiments were completed once. Gel electrophoresis experiments were repeated three times to obtain similar results.
Randomization	Animals were randomly allocated into each group. No specific randomization method was used. For other experiments, we performed side-by-side comparison at the same time to keep the experimental condition uniform. Therefore no randomization is needed.
Blinding	The investigators were not blinded to the data collection as all the assays were run by the same team that performed the animal immunization.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☐	☒ Antibodies
☐	☒ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☐	☒ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☐	☒ Flow cytometry
☒	☐ MRI-based neuroimaging

Antibodies

Antibodies used	anti-RBD Fc chimeric mAb (Cat: 40150-D001, Sino Biological) Clone #D001 PE-anti-human IgG Fc (Cat: 410707, Biolgend) Clone M1310G05 HRP-conjugated goat anti-mouse IgG Ab (Cat: 31430, Invitrogen) Polyclonal Anti-VZV gE protein antibody (Cat: 272686, Abcam) Clone #9 Goat Anti-Mouse IgG H&L (PE)) (Cat: 97024, Abcam) Polyclonal Goat Anti-Mouse IgG Fc (HRP) (Cat: 97265, Abcam) Polyclonal
Validation	anti-RBD Fc chimeric mAb: Du L, et al. (2009) The spike protein of SARS-CoV--a target for vaccine and therapeutic development. Nat Rev Microbiol. 7 (3): 226-36. Anti-VZV gE protein antibody: Wu S et al. Transcriptome Analysis Reveals the Role of Cellular Calcium Disorder in Varicella Zoster Virus-Induced Post-Herpetic Neuralgia. Front Mol Neurosci 14:665931 (2021).

Eukaryotic cell lines

Policy information about cell lines and Sex and Gender in Research

Cell line source(s)	HEK-293 cell line from ATCC (Cat# CRL-1573™) was used.
Authentication	Cell line was not authenticated.
Mycoplasma contamination	The cells were tested negative for mycoplasma contamination. MycoBlue Mycoplasma Detector (Vazyme) was used for detection.
Commonly misidentified lines (See ICLAC register)	No commonly misidentified cell lines were used.

Animals and other research organisms

Policy information about studies involving animals; ARRIVE guidelines recommended for reporting animal research, and Sex and Gender in Research

Laboratory animals	C57BL/6 mice (6-8 weeks, female) were used in this study. Mice were maintained on 12 h light:dark cycles with a housing temperature between 20–24 °C and 40-60% humidity.
Wild animals	The study did not involve wild animals.
Reporting on sex	Only female mice were used in this study without specific consideration of the sex impact on the results. Though publications have shown that male and female mice may differ in immune responses to vaccination (PNAS, 2018 Dec 4; 115(49): 12477–12482.). We followed a general practice using female mice in COVID-19 vaccine studies as used in other studies (Nature 586, 567–571 (2020); Cell 182, 1271–1283.e1–e7, September 3, 2020).
Field-collected samples	No field-collected samples were involved in this study.
Ethics oversight	All mice studies were performed in strict accordance with the guidelines set by the Chinese Regulations of Laboratory Animals and Laboratory Animal-Requirements of Environment and Housing Facilities. Animal experiments were carried out in compliance with the approval protocol from the Institutional Animal Care and Use Committee (IACUC) of Shanghai Model Organisms Center, Inc..

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Flow Cytometry

Plots

Confirm that:

☒ The axis labels state the marker and fluorochrome used (e.g. CD4-FITC).

☒ The axis scales are clearly visible. Include numbers along axes only for bottom left plot of group (a 'group' is an analysis of identical markers).

☒ All plots are contour plots with outliers or pseudocolor plots.

☒ A numerical value for number of cells or percentage (with statistics) is provided.

Methodology

Sample preparation	Human embryonic kidney 293 cells (HEK293) (ATCC) were cultured in Dulbecco's Modified Eagle's Medium (DMEM) (Hyclone) containing 10% fetal bovine serum (FBS) (GEMINI) and 1% Penicillin-Streptomycin (Gibco). All cells were cultured at 37 °C in a 5% CO2 condition. For the measurement of protein expression, cells were transfected with mRNA using Lipofectamine MessengerMAX (Thermo

	Scientific). Briefly, a mix of 2 μg mRNA and 6 μL of Lipofectamine reagent was prepared following the manual instructions and then incubated with cells for 24 or 48 hours. For flow cytometric analysis, cells were collected and stained with live/dead cell dye (Fixable Viability Stain 510, BD) for 5 min. After washing, cells were incubated with anti-RBD chimeric mAb (1:100 dilution, Sino Biological) for 30 min, followed by washing and incubation with PE-anti-human IgG Fc (1:100 dilution, Biolgend) for 30 min. Samples were analyzed on BD Canto II (BD Biosciences). Data were processed using FlowJo V10.1 (Tree Star).
Instrument	BD FACSCanto II (Serial # : R33896203261).
Software	Flowjo version 10.1 was used in FACS analysis.
Cell population abundance	After gating the singlet cells, a total of 10,000 cells were collected for each independent assay.
Gating strategy	In our FACS experiments, only homogeneous cells (HEK293) were used for the evaluation of specific protein translation. In this case, only one fluorescent staining was used to assess the intensity. No other unique gating strategy was applied except for the exclusion of doublets and dead cells.

☐ Tick this box to confirm that a figure exemplifying the gating strategy is provided in the Supplementary Information.