BpForms and *BcForms*: Tools for concretely describing non-canonical polymers and complexes to facilitate comprehensive biochemical networks

Paul F. Lang[1,2,3,*], Yassmine Chebaro[1,2,4,*], Xiaoyue Zheng[1,2,*], John A. P. Sekar[1,2], Bilal Shaikh[1,2], Darren A. Natale[5], and Jonathan R. Karr[1,2,**]

[1]Icahn Institute for Data Science and Genomic Technology, Icahn School of Medicine at Mount Sinai, New York, NY 10029, USA
[2]Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, NY 10029, USA
[3]Department of Biochemistry, Oxford University, South Parks Road, Oxford OX1 3QU, UK
[4]Institut de Génétique et de Biologie Moléculaire et Cellulaire, Institut National de la Santé et de la Recherche Médicale, Centre National de la Recherche Scientifique, Université de Strasbourg, 67404, Illkirch, France
[5]Protein Information Resource, Georgetown University Medical Center, Washington, DC 20007, USA

[*]These authors contributed equally to this work
[**]Correspondence: karr@mssm.edu

August 26, 2019

Abstract

Although non-canonical residues, caps, crosslinks, and nicks play an important role in the function of many DNA, RNA, proteins, and complexes, we do not fully understand how networks of non-canonical macromolecules generate behavior. One barrier is our limited formats, such as IUPAC, for abstractly describing macromolecules. To overcome this barrier, we developed *BpForms* and *BcForms*, a toolkit of ontologies, grammars, and software for abstracting the primary structure of polymers and complexes as combinations of residues, caps, crosslinks, and nicks. The toolkit can help quality control, exchange, and integrate information about the primary structure of macromolecules into fine-grained global networks of intracellular biochemistry.

Keywords

format; software; polymer; proteoform; complex; residue; modification; crosslink; fine-grained network; genome-scale network

1. Background

A central goal in biology is to understand how networks of metabolites, DNA, RNA, proteins, and complexes generate behavior. Non-canonical residues, caps, crosslinks, and nicks are essential to these networks. For example, prokaryotic restriction/modification systems use methylation to selectively degrade foreign DNA, tRNA use pseudouridine to translate multiple codons, and signaling networks use phosphorylation to encode information into the states of proteins.

Recent technical advances have enabled detailed information about individual DNA, RNA, and protein modifications. For example, SMRT-seq can identify the locations of DNA methylations with single-nucleotide resolution[1] and mass-spectrometry can identify hundreds of protein modifications.[2] Furthermore, several repositories have compiled extensive data about non-canonical residues and crosslinks in DNA,[3–6] RNA,[7,8] and proteins,[9–14] as well data about the subunit composition and crosslinks of complexes.[12,14–17] Despite this progress, it remains difficult to integrate this information into fine-grained global networks of intracellular biochemistry, in part, because these resources use chemically-ambiguous and incompatible formats. Consequently, we still do not have a holistic understanding of how non-canonical macromolecules help generate behavior.

Whole-cell (WC) models,[18,19] which aim to predict phenotype from genotype by representing all of the biochemical activity in cells, are a promising tool for integrating diverse information about macromolecules into a holistic understanding of cellular behavior. However, it remains challenging to build fine-grained, global biochemical networks, such as WC models, because we have few tools for capturing the structures of non-canonical macromolecules and linking them together into networks. For example, formats such as BioNetGen[20] and the Systems Biology Markup Language (SBML)[21] are cumbersome for modeling post-transcriptional modification because they have limited capabilities to represent the primary structure of RNA.[22,23] Abstractions of the primary structures of macromolecules that can be combined with modeling frameworks such as SBML would provide a significant step toward fine-grained global biochemical networks. Combined with software tools, such abstractions could also facilitate the curation, exchange, and quality control of structural information about macromolecules for a wide range of omics and systems and synthetic biology research.

Currently, several formats have limited abilities to abstract the primary structures of non-canonical polymers and complexes. Molecular formats which represent each atom and bond, such as the International Chemical Identifier (InChI),[24] the PDB format,[25] and the Simplified Molecular-Input Line-Entry System (SMILES),[26] can represent non-canonical residues, caps, crosslinks, and nicks. However, their fine granularity is cumbersome for network-scale research. Omics and systems biology formats, such as BioPAX,[27] the Biological Expression Language (BEL),[28] the MODOMICS nomenclature,[7] the PRO notation,[13] ProForma,[29] and the Synthetic Biology Open Language (SBOL),[30] use abstractions that are conducive to network-scale research. However, these formats have limited abilities to represent non-canonical residues, caps, crosslinks and nicks, and they do not concretely represent the primary structures of macromolecules.

Toward fine-grained global networks of intracellular biochemistry, we developed *BpForms-BcForms*, an open-source toolkit for abstractly representing the primary structure of polymers and complexes. *BpForms* includes extensible alphabets of hundreds of DNA, RNA and protein residues; an ontology of common crosslinks; and a human and machine-readable grammar for combining residues, residue modifications, intra-chain crosslinks, and nicks into polymers. *BcForms* includes a human and machine-readable grammar for combining polymers, small molecules, and inter-chain crosslinks into complexes. Both tools include software for validating descriptions of macromolecules, calculating properties of macromolecules such as their formula, visualizing macromolecules, and exporting macromolecules to molecular formats such as SMILES. Both tools are available as a web application, REST API, command-line program, and Python library.

Here, we describe the toolkit and demonstrate how it can facilitate omics, systems modeling, and synthetic biology. First, we describe the toolkit, including the alphabets of residues, the ontology of crosslinks, the grammars, the software tools, and the user interfaces. Second, we describe how *BpForms* and *BcForms* can be integrated with knowledge about pathways, kinetic models, and genetic designs through formats such as BioPAX, CellML,[31] SBML, and SBOL. Next, we describe the advantages of the toolkit over existing formats for representing polymers and complexes and existing alphabets of residues. Lastly, we present multiple case studies that illustrate how the toolkit can help researchers describe, quality control, exchange, and integrate diverse information about macromolecules into networks. We anticipate that *BpForms* and *BcForms* will help facilitate fine-grained, global networks of cellular biochemistry.

2. Results

2.1. Toolkit for abstracting non-canonical polymers and complexes

The *BpForms*-*BcForms* toolkit includes several interrelated tools for describing, validating, visualizing, and calculating properties of the primary structure of DNA, RNA, proteins, and complexes (Figure 1). Here, we describe the components of the toolkit including the abstractions and grammars for polymers and complexes; the alphabets of residues; the ontology of crosslinks; the software tools for quality controlling, analyzing, and visualizing macromolecules; the protocols for integrating *BpForms* and *BcForms* with formats for network research; and the user interfaces.

Abstract representation of the primary structure of polymers and complexes. *BpForms* represents polymers as a sequence of residues, a set of crosslinks, a set of nicks, and a Boolean indicator of circularity (Figure 2B, D). *BcForms* represents complexes as a set of subunits and a set of crosslinks (Figure 2A, C). Each subunit is represented by its molecular structure and stoichiometry. The structure of each subunit can be described using *BpForms* or SMILES.

Residues. Each residue is represented by its molecular structure, a list of the atoms which can form bonds with preceding and following residues, and a list of the atoms which are displaced by the formation of these bonds (Figure 2E). These lists of atoms are optional to enable the toolkit to represent internal nucleic and amino acids, as well as 3' and 5' caps. The toolkit can also capture metadata and missing information about residues.

Crosslinks. Each crosslink is represented as lists of the atoms which can form a bond between residues and the atoms which are displaced by the formation of these bonds (Figure 2F). The toolkit represents each nick as a tuple of adjacent residues which are not bonded.

Alphabets of residues and ontology of crosslinks. The toolkit uses a hybrid approach to abstract the molecular details of residues and crosslinks from the descriptions of macromolecules. The chemical details of common residues and crosslinks are abstracted into alphabets of residues and an ontology of crosslinks. Users can define additional residues and crosslinks within descriptions of macromolecules or create custom alphabets and ontologies. This hybrid approach standardizes the representation of common residues and crosslinks while enabling the toolkit to represent any residue or crosslink.

Coordinate system. The toolkit uses a structured coordinate system to describe the atoms involved in each inter-residue bond and crosslink. The coordinate of each repeated subunit ranges from one to the stoichiometry of the subunit. The coordinate of each residue is its position within the residue sequence of its parent polymer. The coordinate of each atom is its position within the canonical SMILES ordering of the atoms in its parent residue. Additional File 1.4 contains more information about the coordinate system.

Examples. Boxes 1 and 2 illustrate the toolkit's grammars for describing polymers and complexes, and Figure 2 illustrates the chemical semantics of a homodimer encoded in the grammars. Additional File 1.2 and the *BpForms* and *BcForms* websites provide detailed descriptions of the grammars and additional examples. Additional File 1.3 contains formal descriptions of the grammars.

Alphabets of DNA, RNA, and protein residues. To support a broad range of research, *BpForms* includes the most extensive alphabets of DNA, RNA, and protein residues to date. The DNA alphabet includes 422 deoxyribose nucleotide monophosphates and 3' and 5' caps derived from data about DNA damage and repair from REPAIREtoire,[4] structural data from the Protein Data Bank Chemical Component Dictionary (PDB CCD),[32] and chemoinformatics data from DNAmod.[3] The RNA alphabet includes 378 ribose nucleotide monophosphates and 3' and 5' caps derived from biochemical data from MODOMICS[33] and the RNA Modification Database[8] and structural data from the PDB CCD. The protein alphabet has 1,435 amino acids and carboxy and amino termini derived from biochemical data from RESID[10] and structural data from the PDB CCD. The *BpForms* website contains pages which display the residues in each alphabet. Additional File 1.5 describes how we constructed the alphabets.

Ontology of crosslinks. To abstract the molecular structures of polymers and complexes, the toolkit includes the first ontology of crosslinks. Currently, the ontology contains 36 common crosslinks. We plan to continue to curate additional crosslinks as needed to represent WC models. The *BpForms* website contains a page which displays the crosslinks in the ontology. Additional File 1.6 describes how we constructed the ontology.

Syntactic and semantic validation of descriptions of macromolecules. To help quality control information about macromolecules, the toolkit can verify the syntactic and semantic correctness of macromolecules encoded in *BpForms* and *BcForms*. First, the toolkit can verify that textual descriptions of macromolecules are syntactically consistent with the *BpForms* and *BcForms* grammars and identify any errors. Second, the toolkit can verify that macromolecules represented by *BpForms* and *BcForms* are semantically consistent and identify any errors. For example, the toolkit can identify pairs of adjacent amino acids that cannot form peptide bonds because the first amino acid does not have a carboxy terminus or because the second amino acid does not have an amino terminus. Additional File 1.7 details the semantic validations implemented by the toolkit. We anticipate that these quality controls will help researchers exchange reliable information and assemble this information into high-quality networks.

Analyses of polymers and complexes. The toolkit can calculate several properties of macromolecules such as their primary structure, major protonation and tautomerization states, chemical formula, molecular weight, and charge. We have begun to use these properties to quality control WC models. For example, we are using the chemical formulae to verify that each reaction is element and charge balanced, including reactions that represent transformations of macromolecules such as the post-transcriptional modification of tRNA.

The toolkit can also compare macromolecules to determine their equality or identify differences. We plan to use this feature to implement automated procedures for merging models that share species and reactions.

Molecular and sequence visualizations. To help analyze macromolecules, the toolkit can generate molecular and sequence visualizations of residues, caps, crosslinks, polymers, and complexes.

The molecular visualizations display each atom and bond and use colors to highlight features such as individual residues, inter-residue and crosslink bonds, and the atoms that are displaced by the formation of the inter-residue bonds (Figure S1A–C). The molecular visualizations can also display the coordinate of each residue and atom. The sequence visualizations include interactive tooltips that describe each non-canonical residue, crosslink, and nick (Figure S1D).

Export to other molecular and sequence formats. For compatibility with structural and biochemical research, the toolkit can export *BpForms* and *BcForms*-encoded macromolecules to molecular formats such as InChI, the PDB format, and SMILES. For compatibility with genomics research, the toolkit can also export the canonical sequences of *BcForms*-encoded polymers to the IUPAC/IUBMB format[34] and FASTA documents.[35]

Integration with frameworks for network-scale research. *BpForms* and *BcForms* can facilitate network-scale research through integration with omics and systems and synthetic biology frameworks such as BioPAX, CellML, SBML, and SBOL. Additional File 1.9 illustrates how *BpForms* and *BcForms* can be incorporated into these frameworks.

User interfaces. *BpForms* and *BcForms* each include four user-friendly interfaces: a web application, a REST API, a command-line program, and a Python library.

2.2. Comparison with existing formats and alphabet-like resources

BpForms and *BcForms* are the first abstractions that can represent the primary structure of any DNA, RNA, protein, and complex, including non-canonical residues, caps, crosslinks, nicks, and circularity. The toolkit also contains the most extensive alphabets of DNA, RNA, and protein residues and the first ontology of concrete crosslinks. Furthermore, the toolkit has several innovative features to facilitate research about non-canonical macromolecules: the toolkit includes a novel coordinate system that makes it easy to address specific atoms in macromolecules, the toolkit uses a novel combination of ontologies and inline definitions of residues and crosslinks to standardize the representation of common residues and crosslinks while accommodating any residue or crosslink, and the toolkit includes novel quality controls for abstractions of the primary structures of macromolecules. Taken together, *BpForms* and *BcForms* are well-suited for network research. Here, we summarize how *BpForms* and *BcForms* improve upon several existing resources for abstracting polymers and complexes.

Comparison of *BpForms* with existing formats for polymers. *BpForms* is the first format that can abstract the primary structure of DNA, RNA, and proteins, including non-canonical residues, caps, crosslinks, nicks, and circularity. In contrast, molecular formats such as SMILES do not abstract the structures of polymers, and abstract formats such as ProForma and network formats such as BioPAX do not represent concrete molecular structures. *BpForms* also provides a unique blend of the features of previous molecular and abstract formats: *BpForms* can capture missing information similar to ProForma, *BpForms* is human-readable like other abstract formats, *BpForms* is machine-readable like molecular formats, *BpForms* is composable with network formats such as SBML like molecular formats, and *BpForms* is backward compatible with the IUPAC/IUBMB format like other abstract formats. Additional File 1.11.1 and Table S1 provide a detailed comparison of *BpForms* with several other formats.

Comparison of *BpForms* alphabets with existing databases. The *BpForms* alphabets are the most extensive alphabets of DNA, RNA, and protein residues because they are based on structural, biochemical, and physiological data from several sources. In addition, the *BpForms* alphabets and the PDB CCD are the only alphabets which consistently represent DNA, RNA, and protein residues and which represent the inter-residue bonding sites of each residue, enabling residues to be combined into concrete molecular structures. In contrast, DNAmod, REPAIRtoire, MODOMICS, RESID, and the RNA Modification Database each only represent DNA, RNA, or protein residues; the residues in DNAmod, REPAIRtoire, MODOMICS, and the RNA Modification Database are hard to compose into polymers because they represent nucleobases and nucleosides rather than nucleotides; and DNAmod, REPAIRtoire, MODOMICS, RESID, and the RNA Modification Database do not capture bonding sites. Additional File 1.11.2 and Table S2 provide a detailed comparison of the *BpForms* alphabets with several other resources.

Comparison of the *BpForms* crosslinks ontology with existing resources. Several resources contain information about crosslinks. In particular, the UniProt controlled vocabulary of post-translational modifications includes textual descriptions of over 100 types of crosslinks. In addition, MOD, REPAIRtoire, and RESID indirectly represent crosslinks by representing crosslinked dimers and trimers.

The *BpForms* ontology is the first resource which directly represents the chemical structures of crosslinks, enabling crosslinks to be composed into concrete structures. In contrast, MOD, REPAIRtoire, and RESID represent crosslinks indirectly and the crosslinks in UniProt do not have concrete chemical semantics. Consequently, the crosslinks in MOD, REPAIRtoire, RESID, and UniProt cannot be composed into concrete structures. Additional File 1.11.3 and Table S3 provide a detailed comparison of the *BpForms* crosslinks ontology with these resources.

Comparison of *BcForms* with existing formats for complexes. Despite the importance of complexes, only a few formats can represent complexes. The PDB format is well-suited to capturing the 3-dimensional structures of complexes. BioPAX and SBOL can also capture the subunit composition of complexes.

BcForms is the first format which abstracts the primary structures of complexes including crosslinks. In contrast, the PDB format has limited capabilities to abstract crosslinks, and BioPAX and SBOL have limited abilities to represent stochiometric information and crosslinks. *BcForms* is also the first format which can be composed with formats for networks such as SBML. Additional File 1.11.4 and Table S4 provide a detailed comparison of *BcForms* with several other formats.

2.3. Case studies

We believe that the *BpForms*-*BcForms* toolkit can support a wide range of omics and systems and synthetic biology research. Here, we illustrate how we have used the toolkit to improve the quality of the PRO database of proteoforms; analyze the metabolic cost of tRNA modification in *Escherichia coli*; refine, expand, a compose a model of MAPK signaling with models of other pathways; and identify constraints on designing new strains of *E. coli*.

Proteomics: Quality control of the Protein Ontology. One of the goals of proteomics is to characterize the proteoforms in cells. Toward a comprehensive catalog of proteoforms, the PRO consortium has manually integrated several different types of data into PRO, a database of 8,095 proteoforms. Because the consortium constructs PRO, in part, by hand, automated quality controls could help the consortium identify and correct errors in PRO.

We have used *BpForms* quality control PRO. First, we encoded each entry in PRO into the *BpForms* grammar and used the *BpForms* software to validate each entry. This identified several types of syntactical and semantic errors. For example, we identified annotated processing sites that have invalid coordinates that are greater than the length of the translated sequence of their parent protein.

We also identified modified residues whose structures are inconsistent with the translated sequences of their parent proteins, such as a phosphorylated serine which is annotated at the position of a tyrosine in the translated sequence of its parent. Second, the consortium corrected these errors. These improvements will be published with the next release later this year.

To enable the consortium to continue to use *BpForms* to quality control PRO, we developed a script which automates this analysis. Going forward, the consortium also plans to use *BpForms* and *BcForms* to visualize and export proteoforms to molecular formats such as SMILES.

Systems biology: Analysis of the metabolic cost of prokaryotic tRNA modification. To achieve WC models, we must integrate information about all of the processes in cells and their interactions. Here, we illustrate how *BpForms* can help integrate information about the interaction between the RNA modification and metabolism of *E. coli* and identify gaps in models.

First, we estimated the abundance of each tRNA from the total observed abundance of tRNA[36,37] and the observed relative abundance of each tRNA.[38] Second, we estimated the synthesis rate of each tRNA from the estimated abundance of each tRNA, the observed half-life of $\mathrm{tRNA}^{\mathrm{Asn}}$,[39] and the observed doubling time of *E. coli* in glucose media.[40] Third, we used *BpForms* to analyze the curated modifications of each tRNA.[7] Fourth, we estimated the total synthesis rate of each modification from the synthesis rate and modification of each tRNA (Figure 3).

This analysis revealed that *E. coli* tRNA contain 26 modified residues, and that the five most abundant residues account for 73.8% of all modifications. Next, we tried to use the iML1515 metabolic model,[41] one of the most comprehensive models of cellular metabolism, to analyze the impact of these modifications on metabolism and understand how *E. coli* allocates its limited metabolic resources among these modifications. This analysis revealed that the model only represents one of the modified residues (9U, pseudouridine). Therefore, the model must be expanded to capture the metabolic cost of tRNA modification.

Systems biology: Systematic identification of gaps in the Kholodenko model of MAPK signaling. The Kholodenko model of the eukaryotic MAPK signaling cascade[42] describes how the cascade transduces extracellular signals for growth, differentiation, and survival into the phosphorylation state of MAPK. However, the model does not account for factors such as the cell's nutritional status.

Toward a more holistic model of the cascade, we used *BpForms* to systematically identify gaps in the Kholodenko model and opportunities to merge the model with models of other pathways. First, we obtained an SBML-encoded version of the model. Second, we determined the specific proteins represented by the model. We had to do this manually because Kholodenko did not report this information. Third, we curated the sequences and post-translational modifications of the species represented by the model from UniProt and encoded them into *BpForms* (Figure 4A). Fourth, we embedded these *BpForms* representations into the SBML representation of the model. We believe that the *BpForms* annotations make the model more understandable.

Fifth, we used the *BpForms* annotations to systematically identify missing proteoforms that could help the model better explain how the MAPK pathway transduces signals. Specifically, we used *BpForms* to identify two missing combinations of the individual protein modifications represented by the model and four missing reactions that involve these species (Figure 4B). These additional species and reactions could help the model better capture the kinetics of MAPKK and MAPKKK activation and deactivation and, in turn, better capture how the pathway transduces signals.

Next, we used the *BpForms* annotations to identify opportunities to merge the Kholodenko model with models of other signaling cascades. Specifically, we searched BioModels for other models that represent similar proteoforms. This analysis identified several models that represent EGFR, PI3K, S6K, and the transcriptional outputs of the MAPK pathway that could be composed with the Kholodenko model. Furthermore, this combination of models enabled us to identify emergent combinations of proteoforms that are missing from the individual models (Figure 4C).

Lastly, to identify opportunities to merge the Kholodenko model with a model of metabolism, we used the *BpForms* annotations to systematically identify unbalanced reactions with missing metabolites. This analysis identified four missing species that, if added to the Kholodenko model, would make the model composable with models of metabolism (Figure 4D).

Synthetic biology: Systematic identification of design constraints. A promising way to engineer cells is to combine naturally-occurring parts, such as genes that encode metabolic enzymes, in an accommodating host, such as *E. coli*. However, there are numerous potential barriers to transforming parts into other cells. For example, parts that require post-translational modifications cannot be transformed into cells which cannot synthesize the modifications. Currently, it is difficult to identify such design constraints because we have limited tools to describe the dependencies of parts. Here, we illustrate how *BpForms* can systematically identify potential flaws in the design of a novel strain of *E. coli* due to missing post-translational modification machinery.

First, we used the PDB and *BpForms* to identify all of the modifications that have been observed in *E. coli*. Second, we used the PDB and *BpForms* to identify modifications which have never been observed in *E. coli* and the proteins which contain these modifications. For example, we found that proteins that contain 4-hydroxproline (PDB CCD: HYP), such as collagen (UniProt: P02452), potentially cannot be transformed into *E. coli*. Third, we used the literature to confirm the absence of these modifications from *E. coli*.[43–45] Table S5 lists the most common modifications which could constrain the transformation of proteins into *E. coli*.

Bioengineers could use this information to more reliably modify strains by limiting designs to post-translationally compatible proteins or by co-transforming parts with their requisite post-translational modification machinery. Furthermore, the synthetic biology community could make such information more accessible for learning design rules by incorporating this information into parts repositories such as SynBioHub.[46] This information would enable these repositories to function as dependency management systems for synthetic organisms, analogous to the Advanced Package Tool (APT) for Ubuntu packages.

3. Discussion

3.1. Community adoption as a common toolkit

Realizing the full potential of *BpForms* and *BcForms* as formats for the primary structures of macromolecules will require acceptance by the omics, systems biology, and synthetic biology communities. We have begun to solicit users by submitting the *BpForms* and *BcForms* grammars to the FAIRsharing registry of standards and the EDAM ontology of formats, contributing the alphabets of residues and the ontology of crosslinks to BioPortal, proposing a protocol for using *BpForms* with SBOL, and helping the PRO consortium use *BpForms* to represent proteoforms. To further encourage community adoption, we plan to encourage the developers of central repositories of DNA, RNA, and protein modifications such as MethSMRT,[5] the PDB, and RMBase[6] to export their data in *BpForms* format. We also plan to stimulate discussion among the BioPAX, CellML, and SBML communities about formalizing our integrations of *BpForms* and *BcForms* with their formats. Additionally, we also plan to use the grammars to generate parsers for other languages, such as C^{++}, to help developers incorporate *BpForms* and *BcForms* into software tools.

3.2. Community adoption as standards

Because *BpForms* and *BcForms* aim to help researchers exchange information, we believe that the alphabets of residues, the ontology of crosslinks, and the grammars should ultimately become community standards. To start, we encourage the community to contribute to *BpForms* and *BcForms* via Git pull requests. Going forward, we would like these resources to be governed by the community through an organization such as the Computational Modeling in Biology Network (COMBINE).[47]

3.3. Integrating closed chemical representations with open informatics representations to enable WC models

BpForms and *BcForms* achieve abstract descriptions of macromolecules by combining a closed, defined grammar with open, extensible ontologies of residues and crosslinks. This hybrid approach enables *BpForms* and *BcForms* to integrate diverse data into chemically-concrete descriptions of a wide range of macromolecules. Achieving WC models swimilarly requires integrating heterogeneous data about a wide range of processes from a wide range of methods and sources into physically-concrete kinetic simulations. Consequently, we believe that hybrid open-closed approaches such as *BpForms* and *BcForms* will be essential for WC modeling. For example, we are developing a hybrid methodology that enables chemically-concrete coarse-grained simulations by using fine-grained reactions to describe the chemical semantics of coarse-grained reactions.

3.4. Enabling multiscale models that bridge structural information with networks

We have begun to use *BpForms* and *BcForms* to describe the chemical semantics of the species represented by network models. Going forward, we also plan to use *BpForms* and *BcForms* to help network models capture finer-grained mechanisms that involve combinatorial interactions, such as how methylation impacts transcription factor-DNA binding. To do this, we are developing a generalized rule-based modeling framework which encapsulates properties such as primary structures into species and links these properties to reactions and rate laws. We anticipate that this framework, together with *BpForms* and *BcForms*, will make it easier to build fine-grained kinetic models of complex processes such as transcriptional backtracking, ribosomal queuing, and tmRNA ribosomal rescuing and combine them into WC models.

4. Conclusions

The *BpForms-BcForms* toolkit abstracts the primary structure of polymers and complexes, including non-canonical residues, caps, crosslinks, nicks, and several types of missing information. Furthermore, the toolkit standardizes the representation of common residues and crosslinks while extensibly accommodating any residue and crosslink by supporting both centrally and user-defined abstractions of residues and crosslinks. The toolkit includes the most extensive alphabets of hundreds of DNA, RNA, and protein residues; the first ontology of common crosslinks; an intuitive coordinate system for the subunits, residues, and atoms in macromolecules; the first human and machine-readable grammar for composing residues, caps, crosslinks, and nicks into polymers and complexes; and user-friendly web, REST, command-line and Python interfaces. The toolkit is backward compatible with the IUPAC/IUBMB format to maximize compatibility with existing bioinformatics tools and knowledge. The toolkit can also be integrated with frameworks for network research such as BioPAX, CellML, SBML, and SBOL.

We anticipate that *BpForms* and *BcForms* will be valuable tools for omics, systems biology, and synthetic biology. First, the tools can help researchers precisely communicate information about macromolecules. For example, the tools can help experimentalists communicate observations of proteoforms and help bioinformaticians exchange information among databases of polymers and complexes. Similarly, the tools can make models and genetic designs more understandable by capturing the semantic meaning of the species represented by models and capturing the structures of the parts of synthetic organisms. For example, *BpForms* could describe proteins produced by expanded genetic codes.

The tools can also help quality control information about macromolecules. For example, the tools could help researchers find errors in reconstructed proteoforms such as inconsistencies between the modified and translated sequences, merge duplicate entries in databases of proteoforms, and identify gaps and element imbalances in models.

In addition, *BpForms* and *BcForms* can help researchers integrate structural, epigenomic, transcriptomic, and proteomic information about macromolecules. For example, the tools can help researchers integrate observations of individual protein modifications into descriptions of entire proteoforms. The tools can also help researchers integrate databases of modified proteins into a model of post-translational processing, combine the model with models of other processes to create WC models, and refine the model by identifying missing combinations of protein states. Similarly, the tools can help bioengineers design biochemical networks by identifying parts that must be co-transformed with post-transcriptional and post-translational modification machinery.

5. Methods

We designed *BpForms* and *BcForms* as separate, but interrelated tools, to provide users lightweight tools for the distinct use cases of describing polymers and complexes. We implemented the toolkit using Python, ChemAxon Marvin, Flask-RESTPlus, Lark, Open Babel,[48] YAML Ain't Markup Language, and Zurb Foundation. Additional File 1.10 provides more information about the implementation.

Declarations

Availability of data and materials

The web applications are located at https://bpforms.org and https://bcforms.org, the REST APIs are located at https://bpforms.org/api and https://bcforms.org/api, the command-line programs and Python libraries are available from PyPI, and the code and ontologies are available at https://github.com/KarrLab.

BpForms and *BcForms* are available open-source under the MIT license. Optionally, a license for ChemAxon Marvin is needed to calculate protonation and tautomerization states and generate molecular visualizations. Free licenses are available for academic researchers.

BpForms and *BcForms* are platform independent. The installation of *BpForms* and *BcForms* requires Python 3.6 or higher, Open Babel, and, optionally, ChemAxon Marvin. A Docker image with these dependencies is available at http://dockerhub.com/u/karrlab.

Documentation, including installation instructions, is available at https://docs.karrlab.org. Interactive Jupyter notebook tutorials are available at https://sandbox.karrlab.org.

This article refers to versions 0.0.9 of *BpForms* and 0.0.2 of *BcForms*.

Competing interests

The authors declare that they have no competing interests.

Funding

This work was supported by the National Institutes of Health [grant numbers R35 GM119771, P41 EB023912]; the National Science Foundation [grant number 1649014]; and the Engineering and Physical Sciences Research Council [grant number EP/L016494/1].

Authors' contributions

PFL, YC, XZ, DAN, and JRK built the alphabets of residues and the ontology of crosslinks. XZ, BS, and JRK developed the software. XZ, DAN, and JRK developed the case studies. PFL, YC, JAPC, and JRK wrote the manuscript. All authors read and approved the final manuscript.

Acknowledgements

We thank Chris Myers and Jacob Beal for helpful discussion about integrating *BpForms* with SBOL and Nicola Hawes for help designing Figure 1.

References

1. Plongthongkum, N., Diep, D. H. & Zhang, K. Advances in the profiling of DNA modifications: cytosine methylation and beyond. *Nat. Rev. Genet.* **15,** 647–661 (2014).
2. Toby, T. K., Fornelli, L. & Kelleher, N. L. Progress in top-down proteomics and the analysis of proteoforms. *Annu. Rev. Anal. Chem.* **9,** 499–519 (2016).
3. Sood, A. J., Viner, C. & Hoffman, M. M. DNAmod: the DNA modification database. *J. Cheminform.* **11,** 30 (2019).
4. Milanowska, K., Krwawicz, J., Papaj, G., Kosiński, J., Poleszak, K., Lesiak, J., Osińska, E., Rother, K. & Bujnicki, J. M. REPAIRtoire–a database of DNA repair pathways. *Nucleic Acids Res.* **39,** D788–D792 (2010).
5. Ye, P., Luan, Y., Chen, K., Liu, Y., Xiao, C. & Xie, Z. MethSMRT: an integrative database for DNA N6-methyladenine and N4-methylcytosine generated by single-molecular real-time sequencing. *Nucleic Acids Res.* **45,** D85–D89 (2017).
6. Xuan, J.-J., Sun, W.-J., Lin, P.-H., Zhou, K.-R., Liu, S., Zheng, L.-L., Qu, L.-H. & Yang, J.-H. RMBase v2.0: deciphering the map of RNA modifications from epitranscriptome sequencing data. *Nucleic Acids Res.* **46,** D327–D334 (2017).
7. Boccaletto, P., Machnicka, M. A., Purta, E., Piątkowski, P., Bagiński, B., Wirecki, T. K., de Crécy-Lagard, V., Ross, R., Limbach, P. A., Kotter, A., *et al.* MODOMICS: a database of RNA modification pathways. 2017 update. *Nucleic Acids Res.* **46,** D303–D307 (2017).
8. Cantara, W. A., Crain, P. F., Rozenski, J., McCloskey, J. A., Harris, K. A., Zhang, X., Vendeix, F. A., Fabris, D. & Agris, P. F. The RNA Modification Database, RNAMDB: 2011 update. *Nucleic Acids Res.* **39,** D195–D201 (2010).

9. Montecchi-Palazzi, L., Beavis, R., Binz, P.-A., Chalkley, R. J., Cottrell, J., Creasy, D., Shofstahl, J., Seymour, S. L. & Garavelli, J. S. The PSI-MOD community standard for representation of protein modification data. *Nat. Biotechnol.* **26,** 864–866 (2008).

10. Garavelli, J. S. The RESID Database of Protein Modifications as a resource and annotation tool. *Proteomics* **4,** 1527–1533 (2004).

11. Hornbeck, P. V., Kornhauser, J. M., Latham, V., Murray, B., Nandhikonda, V., Nord, A., Skrzypek, E., Wheeler, T., Zhang, B. & Gnad, F. 15 years of PhosphoSitePlus®: integrating post-translationally modified sites, disease variants and isoforms. *Nucleic Acids Res.* **47,** D433–D441 (2018).

12. Rose, P. W., Bi, C., Bluhm, W. F., Christie, C. H., Dimitropoulos, D., Dutta, S., Green, R. K., Goodsell, D. S., Prlić, A., Quesada, M., *et al.* The RCSB Protein Data Bank: new resources for research and education. *Nucleic Acids Res.* **41,** D475–D482 (2012).

13. Natale, D. A., Arighi, C. N., Blake, J. A., Bona, J., Chen, C., Chen, S.-C., Christie, K. R., Cowart, J., D'Eustachio, P., Diehl, A. D., *et al.* Protein Ontology (PRO): enhancing and scaling up the representation of protein entities. *Nucleic Acids Res.* **45,** D339–D346 (2016).

14. UniProt Consortium *et al.* UniProt: the universal protein knowledgebase. *Nucleic Acids Res.* **45,** D158–D169 (2017).

15. Meldal, B. H. M., Bye-A-Jee, H., Gajdoš, L., Hammerová, Z., Horáčková, A., Melicher, F., Perfetto, L., Pokorný, D., Lopez, M. R., Türková, A., *et al.* Complex Portal 2018: extended content and enhanced visualization tools for macromolecular complexes. *Nucleic Acids Res.* **47,** D550–D558 (2018).

16. Giurgiu, M., Reinhard, J., Brauner, B., Dunger-Kaltenbach, I., Fobo, G., Frishman, G., Montrone, C. & Ruepp, A. CORUM: the comprehensive resource of mammalian protein complexes—2019. *Nucleic Acids Res.* **47,** D559–D563 (2018).

17. Karp, P. D., Billington, R., Caspi, R., Fulcher, C. A., Latendresse, M., Kothari, A., Keseler, I. M., Krummenacker, M., Midford, P. E., Ong, Q., *et al.* The BioCyc collection of microbial genomes and metabolic pathways. *Brief. Bioinform.* (2017).

18. Karr, J. R., Sanghvi, J. C., Macklin, D. N., Gutschow, M. V., Jacobs, J. M., Bolival Jr, B., Assad-Garcia, N., Glass, J. I. & Covert, M. W. A whole-cell computational model predicts phenotype from genotype. *Cell* **150,** 389–401 (2012).

19. Goldberg, A. P., Szigeti, B., Chew, Y. H., Sekar, J. A., Roth, Y. D. & Karr, J. R. Emerging whole-cell modeling principles and methods. *Curr. Opin. Biotechnol.* **51,** 97–102 (2018).

20. Harris, L. A., Hogg, J. S., Tapia, J.-J., Sekar, J. A., Gupta, S., Korsunsky, I., Arora, A., Barua, D., Sheehan, R. P. & Faeder, J. R. BioNetGen 2.2: advances in rule-based modeling. *Bioinformatics* **32,** 3366–3368 (2016).

21. Hucka, M., Bergmann, F. T., Dräger, A., Hoops, S., Keating, S. M., Le Novère, N., Myers, C. J., Olivier, B. G., Sahle, S., Schaff, J. C., *et al.* The Systems Biology Markup Language (SBML): language specification for level 3 version 2 core. *J. Integr. Bioinform.* **15** (2018).

22. Misirli, G., Cavaliere, M., Waites, W., Pocock, M., Madsen, C., Gilfellon, O., Honorato-Zimmer, R., Zuliani, P., Danos, V. & Wipat, A. Annotation of rule-based models with formal semantics to enable creation, analysis, reuse and visualization. *Bioinformatics* **32,** 908–917 (2015).

23. Courtot, M., Juty, N., Knüpfer, C., Waltemath, D., Zhukova, A., Dräger, A., Dumontier, M., Finney, A., Golebiewski, M., Hastings, J., *et al.* Controlled vocabularies and semantics in systems biology. *Mol. Syst. Biol.* **7,** 543 (2011).

24. Heller, S. R., McNaught, A., Pletnev, I., Stein, S. & Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. *J. Cheminform.* **7,** 23 (2015).

25. Westbrook, J. D. & Fitzgerald, P. in *Structural Bioinformatics* (eds Bourne, P. E. & Weissig, H.) 161–179 (Wiley Online Library, 2003).

26. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inform. Comp. Sci.* **28,** 31–36 (1988).

27. Demir, E., Cary, M. P., Paley, S., Fukuda, K., Lemer, C., Vastrik, I., Wu, G., D'eustachio, P., Schaefer, C., Luciano, J., *et al.* The BioPAX community standard for pathway data sharing. *Nat. Biotechnol.* **28,** 935–942 (2010).

28. Fluck, J., Madan, S., Ansari, S., Karki, R., Rastegar-Mojarad, M., Catlett, N. L., Hayes, W., Szostak, J., Hoeng, J., Peitsch, M., *et al.* Training and evaluation corpora for the extraction of causal relationships encoded in biological expression language (BEL). *Database* **2016,** baw113 (2016).

29. LeDuc, R. D., Schwämmle, V., Shortreed, M. R., Cesnik, A. J., Solntsev, S. K., Shaw, J. B., Martin, M. J., Vizcaino, J. A., Alpi, E., Danis, P., *et al.* ProForma: a standard proteoform notation. *J. Proteome Res.* **17,** 1321–1325 (2018).

30. Cox, R. S., Madsen, C., McLaughlin, J. A., Nguyen, T., Roehner, N., Bartley, B., Beal, J., Bissell, M., Choi, K., Clancy, K., *et al.* Synthetic Biology Open Language (SBOL) version 2.2.0. *J. Integr. Bioinform.* **15** (2018).

31. Cuellar, A., Hedley, W., Nelson, M., Lloyd, C., Halstead, M., Bullivant, D., Nickerson, D., Hunter, P. & Nielsen, P. The CellML 1.1 specification. *J. Integr. Bioinform.* **12,** 4–85 (2015).

32. Westbrook, J. D., Shao, C., Feng, Z., Zhuravleva, M., Velankar, S. & Young, J. The Chemical Component Dictionary: complete descriptions of constituent molecules in experimentally determined 3D macromolecules in the Protein Data Bank. *Bioinformatics* **31,** 1274–1278 (2014).

33. Machnicka, M. A., Milanowska, K., Osman Oglou, O., Purta, E., Kurkowska, M., Olchowik, A., Januszewski, W., Kalinowski, S., Dunin-Horkawicz, S., Rother, K. M., *et al.* MODOMICS: a database of RNA modification pathways–2013 update. *Nucleic Acids Res.* **41,** D262–D267 (2012).

34. Leonard, S. A. IUPAC/IUB single-letter codes within nucleic acid and amino acid sequences. *Curr. Protoc. Bioinformatics,* A–1A (2003).

35. Pearson, W. R. Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol.* **183,** 63–98 (1990).

36. Dong, H., Nilsson, L. & Kurland, C. G. Co-variation of tRNA abundance and codon usage in Escherichia coli at different growth rates. *J. Mol. Biol.* **260,** 649–663 (1996).

37. Mackie, G. A. RNase E: at the interface of bacterial RNA processing and decay. *Nat. Rev. Microbiol.* **11,** 45–57 (2013).

38. Wei, Y., Silke, J. R. & Xia, X. An improved estimation of tRNA expression to better elucidate the coevolution between tRNA abundance and codon usage in bacteria. *Sci. Rep.* **9,** 3184 (2019).

39. Bailly, M., Giannouli, S., Blaise, M., Stathopoulos, C., Kern, D. & Becker, H. D. A single tRNA base pair mediates bacterial tRNA-dependent biosynthesis of asparagine. *Nucleic Acids Res.* **34,** 6083–6094 (2006).

40. Woldringh, C., De Jong, M., Van den Berg, W & Koppes, L. Morphological analysis of the division cycle of two Escherichia coli substrains during slow growth. *J. Bacteriol.* **131,** 270–279 (1977).

41. Monk, J. M., Lloyd, C. J., Brunk, E., Mih, N., Sastry, A., King, Z., Takeuchi, R., Nomura, W., Zhang, Z., Mori, H., *et al.* iML1515, a knowledgebase that computes Escherichia coli traits. *Nat. Biotechnol.* **35,** 904–908 (2017).

42. Kholodenko, B. N. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur. J. Biochem.* **267,** 1583–1588 (2000).

43. Pinkas, D. M., Ding, S., Raines, R. T. & Barron, A. E. Tunable, post-translational hydroxylation of collagen domains in Escherichia coli. *ACS Chem. Biol.* **6,** 320–324 (2011).

44. An, B., Kaplan, D. L. & Brodsky, B. Engineered recombinant bacterial collagen as an alternative collagen-based biomaterial for tissue engineering. *Front. Chem.* **2,** 40 (2014).

45. Yi, Y., Sheng, H., Li, Z. & Ye, Q. Biosynthesis of trans-4-hydroxyproline by recombinant strains of Corynebacterium glutamicum and Escherichia coli. *BMC Biotechnol.* **14,** 44 (2014).

46. McLaughlin, J. A., Myers, C. J., Zundel, Z., Mısırlı, G., Zhang, M., Ofiteru, I. D., Goñi Moreno, A. & Wipat, A. SynBioHub: a standards-enabled design repository for synthetic biology. *ACS Synth. Biol.* **7,** 682–688 (2018).

47. Hucka, M., Nickerson, D. P., Bader, G. D., Bergmann, F. T., Cooper, J., Demir, E., Garny, A., Golebiewski, M., Myers, C. J., Schreiber, F., *et al.* Promoting coordinated development of community-based information standards for modeling in biology: the COMBINE initiative. *Front. Bioeng. Biotechnol.* **3,** 19 (2015).

48. O'Boyle, N. M., Guha, R., Willighagen, E. L., Adams, S. E., Alvarsson, J., Bradley, J.-C., Filippov, I. V., Hanson, R. M., Hanwell, M. D., Hutchison, G. R., *et al.* Open data, open source and open standards in chemistry: the Blue Obelisk five years on. *J. Cheminform.* **3,** 37 (2011).

Figure legends and boxes

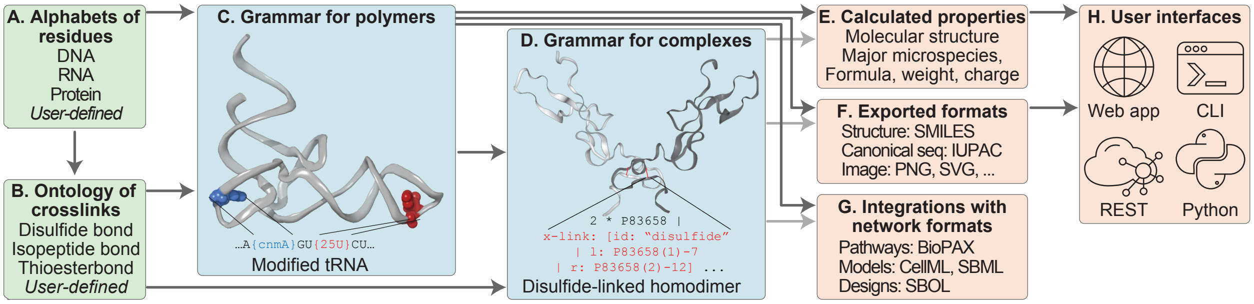
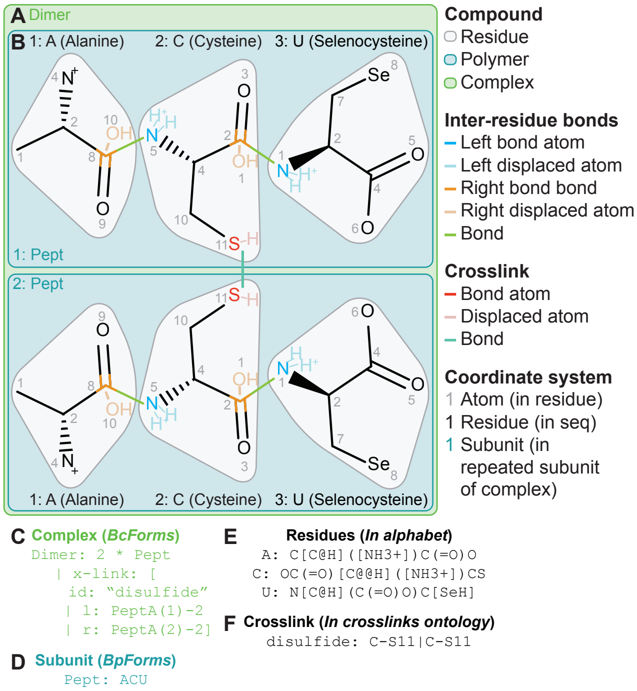


Figure 1. The *BpForms-BcForms* toolkit can abstract, validate, and analyze the primary structures of non-canonical polymers and complexes and help integrate structural information about macromolecules into networks. The toolkit includes (**A**) extensible alphabets that represent individual DNA, RNA and protein residues; (**B**) an ontology of crosslinks; (**C**) a grammar for composing polymers from residues, caps, crosslinks and nicks; (**D**) a grammar for composing complexes from polymers and crosslinks; software tools for validating descriptions of macromolecules, (**E**) calculating molecular properties of macromolecules, (**F**) exporting macromolecules to other formats, and visualizing macromolecules; (**G**) protocols for integrating structural information about macromolecules into omics, systems biology, and synthetic biology formats for networks, models, and genetic designs; and (**H**) multiple user interfaces.



```
Dimer: 2 * Pept
   | x-link: [
     id: “disulfide”
    | l: PeptA(1)-2
    | r: PeptA(2)-2]
```

```
Pept: ACU
```

```
 A: C[C@H]([NH3+])C(=O)O
C: OC(=O)[C@@H]([NH3+])CS
 U: N[C@H](C(=O)O)C[SeH]
```

```
disulfide: C-S11|C-S11
```

Figure 2. ***BpForms* and *BcForms* abstract the primary structures of polymers and complexes as combinations of residues, crosslinks, and nicks.** For example, *BcForms* abstracts a disulfide-linked homodimer (**A**, green box) of a selenocysteine-modified tripeptide (**B**, blue boxes) as two copies of the tripeptide and a single crosslink (**C**, green text) and *BpForms* abstracts the peptide as a sequence of three residues, including selenocysteine (U) (**D**, blue text). These abstractions are enabled by alphabets of residues (**E**, black text) and an ontology of crosslinks (**F**, black text).

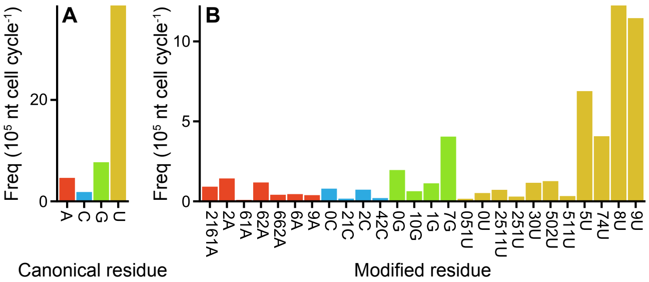


Figure 3. ***BpForms* and *BcForms* can facilitate integrative analyses of fine-grained global intracellular networks.** For example, we used *BpForms* to estimate the metabolic cost of tRNA modification in *E. coli* by canonical residue (**A**) and modified residue (**B**) from information about the modification, abundance, and turnover of each tRNA.

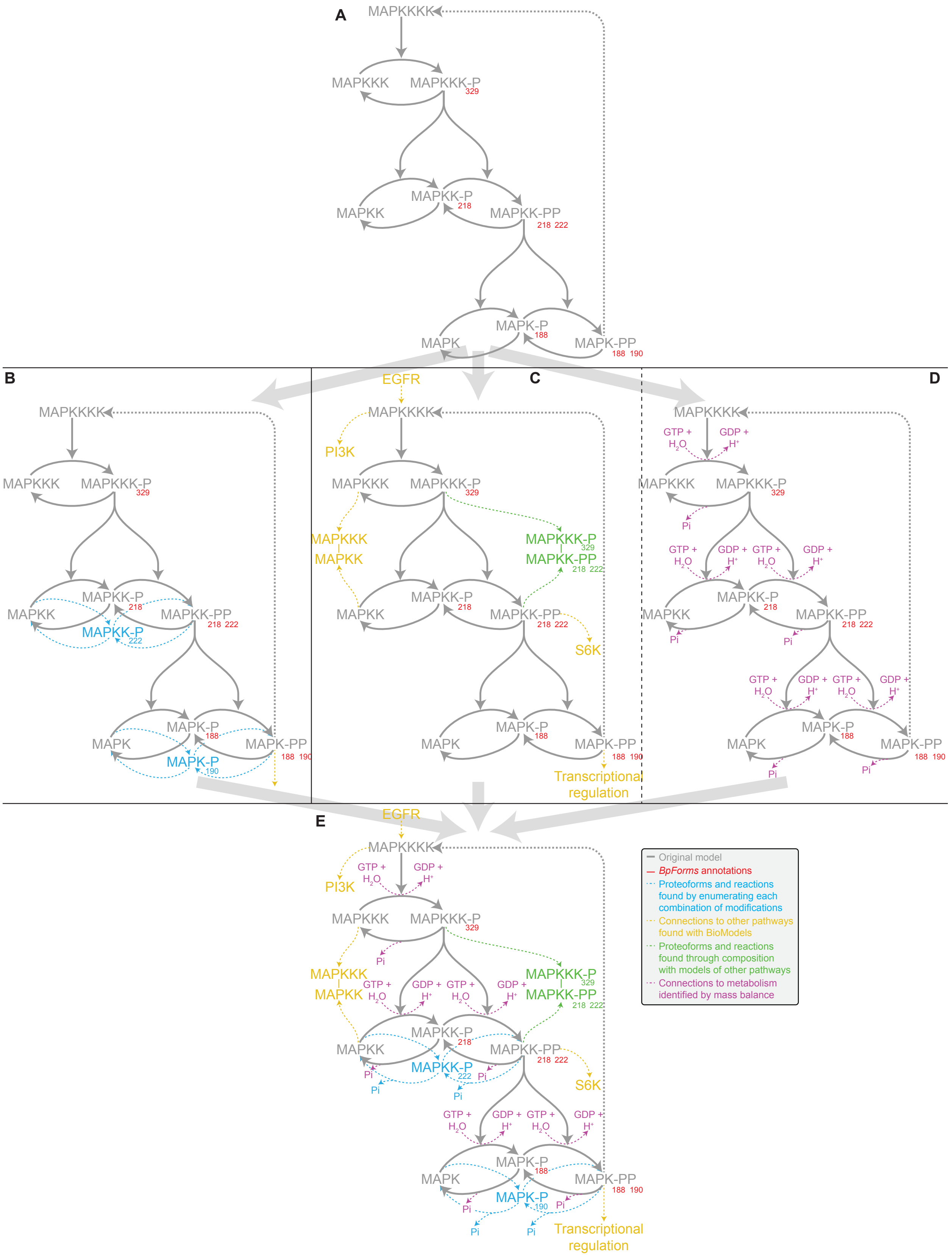

A
MAPKKKK
MAPKKK
MAPKKK-P
329
MAPKK
MAPKK-P
218
MAPKK-PP
218 222
MAPK
MAPK-P
188
MAPK-PP
188 190
B
MAPKKKK
MAPKKK
MAPKKK-P
329
MAPKK
MAPKK-P
218
MAPKK-PP
218 222
MAPKK-P
222
MAPK
MAPK-P
188
MAPK-PP
188 190
MAPK-P
190
C
EGFR
MAPKKKK
PI3K
MAPKKK
MAPKKK-P
329
MAPKKK
MAPKK
MAPKKK-P
329
MAPKK-PP
218 222
MAPKK
MAPKK-P
218
MAPKK-PP
218 222
S6K
MAPK
MAPK-P
188
MAPK-PP
188 190
Transcriptional regulation
D
MAPKKKK
GTP + H2O
GDP + H+
MAPKKK
MAPKKK-P
329
Pi
GTP + H2O
GDP + H+
GTP + H2O
GDP + H+
MAPKK
MAPKK-P
218
MAPKK-PP
218 222
Pi
Pi
GTP + H2O
GDP + H+
GTP + H2O
GDP + H+
MAPK
MAPK-P
188
MAPK-PP
188 190
Pi
Pi
E
EGFR
MAPKKKK
PI3K
GTP + H2O
GDP + H+
MAPKKK
MAPKKK-P
329
Pi
MAPKKK
MAPKK
GTP + H2O
GDP + H+
GTP + H2O
GDP + H+
MAPKKK-P
329
MAPKK-PP
218 222
MAPKK
MAPKK-P
218
MAPKK-PP
218 222
MAPKK-P
222
Pi
Pi
Pi
Pi
S6K
GTP + H2O
GDP + H+
GTP + H2O
GDP + H+
MAPK
MAPK-P
188
MAPK-PP
188 190
MAPK-P
190
Pi
Pi
Pi
Pi
Transcriptional regulation
Original model
BpForms annotations
Proteoforms and reactions found by enumerating each combination of modifications
Connections to other pathways found with BioModels
Proteoforms and reactions found through composition with models of other pathways
Connections to metabolism identified by mass balance

Figure 4. ***BpForms* and *BcForms* can facilitate the construction, expansion, composition, and refinement of fine-grained global intracellular networks.** For example, we used *BpForms* to systematically identify ways to improve and expand the Kholodenko model of MAPK signaling (**A**, grey) by using *BpForms* to capture the semantic meaning of each species (**A**, red), identify missing protein states (**B**, blue), identify other models that represent similar proteins which could be composed with the Kholodenko model (**C**, yellow) which could reveal additional missing combinations of species (**C**, green), and identify mass imbalances which indicate missing metabolites which could facilitate composition with metabolic models (**D**). Together, this could enable a substantially expanded model (**E**).

Residue sequence

This example illustrates how to use *BpForms* to describe a DNA which begins with deoxyinosine.

```
{dI}ACGC
```

User-defined residues

Residues which are not captured by our public alphabets can be captured within descriptions of polymers. This example illustrates how to describe a protein which ends with N^5-methyl-L-arginine.

```
CRGN[
     id:  "AA0305"
   | structure:  "OC(=O)[C@H](CCCN(C(=[NH2])N)C)[NH3+]"
   | l-bond-atom:  N16-1
   | r-bond-atom:  C2
   | l-displaced-atom:  H16+1
   | l-displaced-atom:  H16
   | r-displaced-atom:  O1
   | r-displaced-atom:  H1
   | name:  "N5-methyl-L-arginine"
   | synonym:  "delta-N-methylarginine"
   | synonym:  "N5-carbamimidoyl-N5-methyl-L-ornithine"
   | identifier:  "MOD:00310" @ "mod"
   | identifier:  "CHEBI:21848" @ "chebi"
   | base-monomer:  "R"
   | comments:  "Generated by protein-arginine N5-methyltransferase (EC 2.1.1.-)."
  ]
```

Crosslinks and nicks

This example illustrates how to describe a peptide that contains a disulfide bond between the cysteines at the first and third positions and a nick between the cysteine and alanine at the first and second positions.

```
C:AC | x-link:  [
     id:  "disulfide"
   | l:  1 | r:  3
  ]
```

User-defined crosslinks

Crosslinks which are not captured by our public ontology can be described inline. This example illustrates how to describe a peptide that contains a disulfide bond between the cysteines at the first and third positions.

```
CAC | x-link:  [
     l-bond-atom:  1S11 | r-bond-atom:  3S11
   | l-displaced-atom:  1H11 | r-displaced-atom:  3H11
   | comments:  "disulfide bond between 1C and 3C"
  ]
```

Circularity

This example illustrates how to describe a circular di-deoxyribonucleic acid.

```
AC | circular
```

Missing knowledge

User-defined residues can also capture missing information about the mass, charge, location, and biosynthesis of residues. This example illustrates how to describe a protein which contains a methylated cysteine or asparagine at an unknown position between the fifth and tenth residues.

```
CRGN[
     base-monomer:  "C"
   | delta-mass:  12 | delta-charge:  0
   | position:  5-10 [C, N]
  ]
EGYNNYCRAKYRGH
```

Box 1. Examples of the *BpForms* grammar for describing polymers.

Subunit composition

This example illustrates how to use *BcForms* to describe MalEFGK (Complex Portal: CPX-1932), a heteropentameric maltose ABC transporter.

```
MalE + MalF + MalG + 2 * MalK
```

Crosslinks

This example illustrates how to use the crosslinks ontology to describe a disulfide-linked antiparallel homodimer of disintegrin schistatin of *Echis carinatus* (UniProt: P83658).

```
2 * P83658
  | x-link:  [
        id:  "disulfide"
      | l:  P83658(1)-7
      | r:  P83658(2)-12
    ]
  | x-link:  [
        id:  "disulfide"
      | l:  P83658(1)-12
      | r:  P83658(2)-7
    ]
```

User-defined crosslinks

Crosslinks which are not captured by our public ontology can be defined within descriptions of complexes. This example illustrates how to describe the crosslinking of 10 kDa chaperonin (UniProt: P9WPE5) of *Mycobacterium tuberculosis* with prokaryotic ubiquitin-like protein Pup (UniProt: P9WHN5) via a isoglutamyl lysine isopeptide bond (RESID: AA0124). Cells use this crosslink to mark 10 kDa chaperonin for proteasomal degradation.

```
P9WPE5 + P9WHN5
  | x-link:
   [
      l-bond-atom:  P9WHN5(1)-100N1-1
    | r-bond-atom:  P9WPE5(1)-63C2
    | l-displaced-atom:  P9WHN5(1)-100H1+1
    | l-displaced-atom:  P9WHN5(1)-100H1
    | r-displaced-atom:  P9WPE5(1)-63N1
    | r-displaced-atom:  P9WPE5(1)-63H1
    | r-displaced-atom:  P9WPE5(1)-63H1
    | comments:  "isoglutamyl lysine isopeptide bond"
   ]
```

Box 2. Examples of the *BcForms* grammar for describing complexes.

BpForms and *BcForms*: Additional File 1, Figure S1, and Tables S1-S5

Paul F. Lang[1,2,3,*], Yassmine Chebaro[1,2,4,*], Xiaoyue Zheng[1,2,*], John A. P. Sekar[1,2], Bilal Shaikh[1,2], Darren A. Natale[5], and Jonathan R. Karr[1,2,**]

[1]Icahn Institute, Icahn School of Medicine at Mount Sinai, US
[2]Department of Genetics & Genomic Sciences, Icahn School of Medicine at Mount Sinai, US
[3]Department of Biochemistry, Oxford University, UK
[4]Institut de Génétique et de Biologie Moléculaire et Cellulaire, FR
[5]Protein Information Resource, Georgetown University Medical Center, US

[*]These authors contributed equally to this work
[**]Correspondence: karr@mssm.edu

Contents

1. Features of the *BpForms-BcForms* toolkit

The toolkit has the following features:

- **Concrete:** To help researchers communicate and integrate data about macromolecules, the grammars can capture the primary structures of macromolecules, including non-canonical residues, caps, crosslinks, and nicks.
- **Abstract:** To facilitate network-scale research, *BpForms* and *BcForms* use alphabets of residues and an ontology of crosslinks to abstract the structures of polymers and complexes.
- **Extensible:** To capture any polymer or complex, users can define residues and complexes inline or define custom alphabets and ontologies.
- **Structured coordinates:** To compose residues and crosslinks into polymers and complexes, each subunit, residue, and atom has a unique coordinate relative to its parent.
- **Context-free:** To help integrate information about the processes which synthesize and modify macromolecules, the grammars capture the structures of macromolecules separately from the processes which generate macromolecules.
- **User-friendly:** To ensure *BpForms* and *BcForms* is easy to use, the grammars are human-readable, and the toolkit includes web applications and command-line programs.
- **Machine-readable:** The grammars are machine-readable to enable analyses of macromolecules.
- **Composable:** To facilitate network-scale research, we have developed protocols for composing the grammars with formats such as BioPAX, CellML, SBML, and SBOL.
- **Backward-compatible:** *BpForms* is backward compatible with the IUPAC/IUBMB format to maximize compatibility with existing formats, software, and knowledge.

2. Overview of the grammars for polymers and complexes

2.1. *BpForms* grammar for polymers

BpForms describes polymers using a similar grammar to the IUPAC/IUBMB format. For example, `GaTC` describes a DNA 4-mer which contains 6-methyladenine (a) at the second residue. `RCAC | x-link: [type: "disulfide" | l: 2 | r: 4]` describes a protein 4-mer which contains a disulfide bond between the cysteines at the second and fourth positions. `AUUCG | circular` describes a circular RNA 5-mer.

Here, we summarize the grammar. Box 1 and https://bpforms.org contain several examples of polymers encoded in the grammar. Section 3.1 contains a formal description of the grammar.

Residue sequence

Residues that belong to alphabets can be indicated by their codes. Residues which have single-character codes can be indicated by their codes. For example, `A` describes deoxyadenosine monophosphate. Residues which have multiple-character codes can be indicated by enclosing their codes in brackets. For example, `{m2C}` describes 2-O-methylcytidine monophosphate.

Residues which are not in the alphabets can be described in three ways: users can submit pull requests to add residues to the public alphabets, define their own alphabets, or define residues inline within descriptions of polymers.

User-defined residues. Residues can be defined inline as a pipe-separated set of attribute-value pairs enclosed in brackets. For example, `[name: "m6G" | ...]` describes 6-O-methylguanosine monophosphate (m^{2}G). The `structure` attribute can capture the molecular structure of the residue in the SMILES format. For example, `structure: "COc1nc(N)nc2c1ncn2C1CC(C(O1)COP(=O)([-])[O-])O"` represents the structure of m^{2}G. Optionally, the `l-bond-atom` and `r-bond-atom` attributes can capture the atoms which can form bonds with preceding (l, left) and following (r, right) residues, and the `l-displaced-atom` and `r-displaced-atom` attributes can capture the atoms which are displaced by the formation of these bonds. The values of these attributes indicate the element, coordinate, and change in the formal charge of each atom upon bonding an adjacent residue. For example, `l-bond-atom: P20` indicates the phosphorous of m^{6}G that forms bonds with preceding residues and `l-displaced-atom: O23-1` indicates the oxygen which is displaced by the formation of these bonds.

Several optional attributes can capture metadata about residues. The `id`, `name`, and `synonym` attributes can capture labels. The `identifier` attribute can capture references to equivalent entries in databases. For example, `identifier: "6-O-methylguanine" @ "dnamod"` indicates a reference to an entry in DNAmod. The `base-monomer` attribute can indicate how residues are synthesized from other residues. For example, `base-monomer: "G"` indicates that m^{6}G is derived from guanosine (G). The `comments` attribute can capture additional information about residues.

Crosslinks

BpForms represents each crosslink between two residues as (a) a pair of the atoms which form a bond between the residues and (b) a set of the atoms which are displaced by the formation of the bond. Crosslinks which belong to the ontology of crosslinks can be described by the `x-link` keyword followed by a pipe-separated list of attribute-value pairs enclosed in brackets. The `type` attribute indicates the type of the crosslink. The value of this attribute must refer to an entry in the crosslinks ontology. The `l` and `r` attributes indicate the coordinates of the residues involved in the crosslink. For example, `x-link: [type: "disulfide" | l: 2 | r: 5]` indicates a disulfide bond between cysteines at the second and fifth residues.

Users can also define crosslinks by submitting pull requests to add crosslinks to the public ontology, defining their own ontology, or defining crosslinks inline within descriptions of polymers.

User-defined crosslinks. Crosslinks can be defined inline as a pipe-separated set of attribute-value pairs enclosed in brackets. Similar to user-defined residues, the `l-bond-atom` and `r-bond-atom` attributes describe the atoms which form covalent bonds and the `l-displaced-atom` and `r-displaced-atom` attributes describe the atoms which are displaced by the formation of these bonds. For example, `l-bond-atom: 2O11-1 | r-bond-atom: 7P20` indicates that the crosslink involves a covalent bond between the oxygen at the eleventh position of the second residue and the phosphorous at the twentieth position of the seventh residue, and that the formation of the crosslink decreases the formal charge of the oxygen by one electron. The `order` attribute can capture the order (single, double, triple, or aromatic) of the bond. The `stereo` attribute can capture the stereochemistry (wedge, hash, up, or down) of the bond. The `comments` attribute can capture additional textual information about the crosslink.

Nicks

Nicks can be defined by inserting a colon between the residues involved in the nick. For example, `AC:DE` describes a nick between the second and third residues of a peptide.

Linear or circular topology

Optionally, the `circular` attribute can describe a bond between the left bonding site of the first residue and the right bonding site of the last residue. For example, `CTAC | circular` describes a circular DNA tetramer.

Missing knowledge

User-defined residues can also capture four types of uncertainty about polymers. The `delta-mass` and `delta-charge` attributes can describe mass and charge which have been observed, but which cannot be assigned to an exact molecular structure. For example, `[id: "R" | delta-mass: 17 | delta-charge: 0]` indicates a residue whose mass is 17 Da greater than that of arginine, but whose exact structure is not known. The `position` attribute can capture uncertainty about the location and biosynthesis of a non-canonical residue. For example, `AC[base-monomer: "Y" | identifier: "MOD:00696" @ "mod" | position: 3-5 [S, T, Y]]ST` indicates a peptide that contains a phosphorylated serine, threonine, or tyrosine between positions five and ten.

2.2. *BcForms* grammar for complexes

BcForms describes complexes using a grammar that is similar to a linear mathematical expression. For example, `CHAF1A + SUMO1 | x-link: [ ...]` describes a crosslinked heterodimer of chromatin assembly factor 1 subunit A (CHAF1A) and small ubiquitin-related modifier 1 (SUMO1).

Here, we summarize the grammar. Box 2 and https://bcforms.org contain examples of complexes encoded in the grammar. Section 3.2 contains a formal description of the grammar.

Subunit composition

The subunits involved in complexes and their stoichiometries can be described as a linear expression. For example, `2 * HBA1 + 2 * HBB` describes hemoglobin HbA, a heterotetramer composed of two subunits of HBA1 (UniProt: P69905) and two subunits of HBB (UniProt: P68871). Each subunit can be represented using *BpForms* or SMILES.

Intersubunit crosslinks

BcForms captures crosslinks similar to *BpForms*. Crosslinks which belong to the ontology can be described using the coordinates of the residues involved in the crosslink. For example, `x-link: [type: "disulfide" | l: P83658(1)-7 | r: P83658(2)-12 | ...]` describes a disulfide bond between the seventh and twelfth cysteines of two subunits of disintegrin schistatin (UniProt: P83658). The `l` and `r` attributes describe the subunit type, subunit coordinate, and residue coordinate of the atoms involved in the crosslink. Users can also define crosslinks inline similarly to *BpForms*. For example, `x-link: [l-bond-atom: P83658(1)-7S11 | r-bond-atom: P83658(2)-12S11 | ...]` describes the same disulfide bond between the seventh and twelfth cysteines of disintegrin schistatin.

Complexes can have zero, one, or more crosslinks. Each crosslink can involve the formation of one or more covalent bonds and the displacement of zero or more atoms.

3. Formal descriptions of the grammars for polymers and complexes

The *BpForms* and *BcForms* grammars are defined in Extended Backus-Naur Form (EBNF) [1] using Lark [2]. The grammars are available at https://github.com/KarrLab. Below are descriptions of the grammars in Backus-Naur Form (BNF).

3.1. *BpForms* grammar for polymers

BpForm

⟨bpform⟩ ⊨ ⟨seq⟩ ⟨x-links⟩ ⟨circularity⟩

Sequence of residues and nicks

⟨seq⟩ ⊨ ⟨residue⟩ | ⟨residue⟩ ⟨seq⟩ | *nick* ⟨residue⟩ ⟨seq⟩

⟨residue⟩ ⊨ ⟨single-code-residue⟩ | ⟨delimited-multi-code-residue⟩ | ⟨user-residue⟩

Alphabet-defined residues

⟨single-code-residue⟩ ⊨ ⟨code⟩

⟨delimited-multi-code-residue⟩ ⊨ `{` ⟨multi-code⟩ `}`

⟨code⟩ ⊨ *non-whitespace character*

⟨multi-code⟩ ⊨ ⟨code⟩ | ⟨code⟩ ⟨multi-code⟩

User-defined residues

⟨user-residue⟩ ⊨ `[` ⟨attrs⟩ `]`

⟨attrs⟩ ⊨ ⟨attr⟩ | ⟨attr⟩ `` `|' `` ⟨attrs⟩ | λ

⟨attr⟩ ⊨ ⟨id⟩ | ⟨name⟩ | ⟨synonym⟩ | ⟨identifier⟩ | ⟨structure⟩ | ⟨atom⟩ | ⟨base⟩ | ⟨delta-mass⟩ | ⟨delta-charge⟩ | ⟨position⟩ | ⟨comments⟩

⟨id⟩ ⊨ `id` : " ⟨escaped-string⟩ "

⟨name⟩ ⊨ `name` : " ⟨escaped-string⟩ "

⟨synonym⟩ ⊨ `synonym` : " ⟨escaped-string⟩ "

⟨identifier⟩ ⊨ `identifier` : " ⟨identifier-ns⟩ " `@` " ⟨identifier-id⟩ "

⟨identifier-ns⟩ ⊨ ⟨escaped-string⟩

⟨identifier-id⟩ ⊨ ⟨escaped-string⟩

⟨structure⟩ ⊨ `structure` : " ⟨string⟩ "

⟨atom⟩ ⊨ ⟨atom-type⟩ : ⟨atom-element⟩ ⟨atom-index⟩ ⟨atom-charge⟩

⟨base⟩ ⊨ `base-monomer` : " ⟨multi-code⟩ "

⟨delta-mass⟩ ⊨ `delta-mass` : ⟨number⟩

⟨delta-charge⟩ ⊨ `delta-charge` : ⟨integer⟩

⟨position⟩ ⊨ `position` : ⟨position-start⟩ `-` ⟨position-end⟩ ⟨position-residues⟩

⟨position-start⟩ ⊨ ⟨positive-integer⟩

⟨position-end⟩ ⊨ ⟨positive-integer⟩

⟨position-residues⟩ ⊨ `[` ⟨position-residue-codes⟩ `]` | λ

⟨position-residue-codes⟩ ⊨ ⟨multi-code⟩ | ⟨multi-code⟩ `` `|' `` ⟨position-residue-codes⟩

⟨comments⟩ ⊨ `comments` : " ⟨escaped-string⟩ "

```
                    Crosslinks
                   ⟨x-links⟩  ⊨  ⟨x-link⟩ | ⟨x-link⟩ ⟨x-links⟩ | λ
                    ⟨x-link⟩  ⊨  ‘|’ x-link : [ ⟨x-link-attrs⟩ ]
              ⟨x-link-attrs⟩  ⊨  ⟨onto-x-link-attrs⟩ | ⟨user-x-link-attrs⟩
         ⟨onto-x-link-attrs⟩  ⊨  ⟨onto-x-link-attr⟩ | ⟨onto-x-link-attr⟩‘|’
                                 ⟨onto-x-link-attrs⟩
          ⟨onto-x-link-attr⟩  ⊨  ⟨onto-x-link-type⟩ | ⟨onto-x-link-l-monomer⟩ |
                                 ⟨onto-x-link-r-monomer⟩
          ⟨onto-x-link-type⟩  ⊨  type : " ⟨non-whitespace-characters⟩ "
     ⟨onto-x-link-l-monomer⟩  ⊨  l : ⟨positive-integer⟩
     ⟨onto-x-link-r-monomer⟩  ⊨  r : ⟨positive-integer⟩

        User-defined crosslinks
         ⟨user-x-link-attrs⟩  ⊨  ⟨user-x-link-attr⟩ | ⟨user-x-link-attr⟩ ‘|’
                                 ⟨user-x-link-attrs⟩ | λ
          ⟨user-x-link-attr⟩  ⊨  ⟨user-x-link-atom⟩ | ⟨user-x-link-order-attr⟩ |
                                 ⟨user-x-link-stereo-attr⟩ | ⟨user-x-link-comments-attr⟩
          ⟨user-x-link-atom⟩  ⊨  ⟨atom-type⟩ ⟨atom-residue⟩ ⟨atom-element⟩ ⟨atom-index⟩
                                 ⟨atom-charge⟩
    ⟨user-x-link-order-attr⟩  ⊨  order : "⟨user-x-link-order⟩ "
         ⟨user-x-link-order⟩  ⊨  single | double | triple | aromatic
   ⟨user-x-link-stereo-attr⟩  ⊨  stereo : "⟨user-x-link-stereo⟩ "
        ⟨user-x-link-stereo⟩  ⊨  wedge | hash | up | down
 ⟨user-x-link-comments-attr⟩  ⊨  commments : " ⟨escaped-string⟩ "

                         nicks
                      ⟨nick⟩  ⊨  :

                   Circularity
               ⟨circularity⟩  ⊨  ‘|’ circular | λ

            User-defined atoms
                 ⟨atom-type⟩  ⊨  l-bond-atom | l-displaced-atom |
                                 r-bond-atom | r-displaced-atom
              ⟨atom-residue⟩  ⊨  ⟨positive-integer⟩
              ⟨atom-element⟩  ⊨  A...Z | A...Z a...z
                ⟨atom-index⟩  ⊨  ⟨positive-integer⟩
               ⟨atom-charge⟩  ⊨  ⟨sign⟩ ⟨non-negative-integer⟩ | λ
                      ⟨sign⟩  ⊨  + | -

                    Primitives
```

⟨string⟩ ⊨ *string*

⟨escaped-string⟩ ⊨ *quote escaped string*

⟨non-whitespace-characters⟩ ⊨ *non-whitespace characters*

⟨integer⟩ ⊨ *integer*

⟨positive-integer⟩ ⊨ *positive integer*

⟨non-negative-integer⟩ ⊨ *non-negative integer*

3.2. *BcForms* grammar for complexes

BcForm

⟨bcform⟩ ⊨ ⟨subunits⟩ ⟨x-links⟩

Subunits

⟨subunits⟩ ⊨ ⟨subunit⟩ | ⟨subunit⟩ + ⟨subunits⟩

⟨subunit⟩ ⊨ ⟨coefficient⟩ * ⟨id⟩ | ⟨id⟩

⟨id⟩ ⊨ ⟨non-white space characters⟩

⟨coefficient⟩ ⊨ ⟨positive integer⟩

Crosslinks

⟨x-links⟩ ⊨ ⟨x-link⟩ | ⟨x-link⟩ ⟨x-links⟩

⟨x-link⟩ ⊨ ‘|’ crosslink : [⟨x-link-attrs⟩]

⟨x-link-attrs⟩ ⊨ ⟨onto-x-link-attrs⟩ | ⟨user-x-link-attrs⟩

⟨onto-x-link-attrs⟩ ⊨ ⟨onto-x-link-attr⟩ | ⟨onto-x-link-attr⟩ ‘|’ ⟨onto-x-link-attrs⟩

⟨onto-x-link-attr⟩ ⊨ ⟨onto-x-link-type⟩ | ⟨onto-x-link-l-monomer⟩ | ⟨onto-x-link-r-monomer⟩

⟨onto-x-link-type⟩ ⊨ type : " ⟨onto-x-link-type-value⟩ "

⟨onto-x-link-type-value⟩ ⊨ ⟨non-whitespace characters⟩

⟨onto-x-link-l-monomer⟩ ⊨ l : ⟨atom-subunit-id⟩ (⟨atom-subunit-index⟩) - ⟨atom-monomer-index⟩

⟨onto-x-link-r-monomer⟩ ⊨ r : ⟨atom-subunit-id⟩ (⟨atom-subunit-index⟩) - ⟨atom-monomer-index⟩

User-defined crosslinks

⟨user-x-link-attrs⟩ ⊨ ⟨user-x-link-attr⟩ | ⟨user-x-link-attr⟩ ‘|’ ⟨user-x-link-attrs⟩

⟨user-x-link-attr⟩ ⊨ ⟨user-x-link-atom⟩ | | ⟨user-x-link-order-attr⟩ | ⟨user-x-link-stereo-attr⟩ | ⟨user-x-link-comments-attr⟩

⟨user-x-link-order-attr⟩ ⊨ order : "⟨user-x-link-order⟩ "

⟨user-x-link-order⟩ ⊨ single | double | triple | aromatic

⟨user-x-link-stereo-attr⟩	⊨	`stereo : "`⟨user-x-link-stereo⟩ `"`
⟨user-x-link-stereo⟩	⊨	`wedge` | `hash` | `up` | `down`
⟨user-x-link-comments-attr⟩	⊨	`commments : "` ⟨escaped-string⟩ `"`
User-defined atoms		
⟨user-x-link-atom⟩	⊨	⟨atom-type⟩ `:` ⟨atom-subunit-id⟩ `(` ⟨atom-subunit-index⟩ `)` `-` ⟨atom-monomer-index⟩ ⟨atom-index⟩ ⟨atom-element⟩ ⟨atom-charge⟩
⟨atom-type⟩	⊨	`l-bond-atom` | `r-bond-atom` | `l-displaced-atom` | `r-displaced-atom`
⟨atom-subunit-id⟩	⊨	⟨non-whitespace characters⟩
⟨atom-subunit-index⟩	⊨	⟨positive integer⟩
⟨atom-monomer-index⟩	⊨	⟨positive integer⟩
⟨atom-index⟩	⊨	⟨positive integer⟩
⟨atom-element⟩	⊨	`A...Z` | `A...Z a...z`
⟨atom-charge⟩	⊨	⟨sign⟩ ⟨atom-charge-value⟩ | λ
⟨sign⟩	⊨	`+` | `-`
⟨atom-charge-value⟩	⊨	⟨non-negative integer⟩
Primitives		
⟨escaped-string⟩	⊨	*quote escaped string*
⟨non-whitespace-characters⟩	⊨	*non-whitespace characters*
⟨positive-integer⟩	⊨	*positive integer*
⟨non-negative-integer⟩	⊨	*non-negative integer*

4. Coordinate system

To facilitate descriptions of crosslinks, each residue, and atom represented by *BpForms* has a unique coordinate (Figure 2). The coordinate of each residue is its position within the residue sequence of its parent polymer. The coordinate of each atom is a tuple of the coordinate of its parent residue and its position within the canonical SMILES ordering of the atoms in its parent residue prior to incorporation into polymers.

Each subunit, residue, and atom represented by *BcForms* also has a unique coordinate (Figure 2). The coordinates of repeated subunits range from one to the stoichiometry of the subunit. The coordinate of each residue is a two-tuple of the coordinate of its parent subunit and its position within the residue sequence of its parent subunit. The coordinate of each atom is a three-tuple of the coordinate of its parent subunit, the position of its parent residue within the residue sequence of its parent polymer, and its position within the canonical SMILES ordering of its parent residue.

5. Construction of the alphabets of DNA, RNA, and protein residues

To support a broad range of research, we developed the alphabets of DNA, RNA, and protein residues by merging residues from multiple databases. We developed the DNA alphabet by combining the deoxyribose nucleotide monophosphates and 3' and 5' DNA caps from the PDB Chemical Component Dictionary (PDB CCD) [3] with the verified DNA nucleobases from DNAmod [4] and the deoxyribose nucleosides from REPAIRtoire [5] that had concrete structures. We developed the RNA alphabet by combining the ribose nucleotide monophosphates and 3' and 5' RNA caps from the PDB CCD with the ribose nucleosides from MODOMICS [6] and the RNA Modification Database [7] that had concrete structures. We developed the protein alphabet by merging residues from the PDB CCD and RESID [8].

First, we downloaded, scraped, and manually extracted residues from DNAmod, MODOICS, PDB CCD, REPAIRtoire, RESID, the RNA Modification Database. Second, we parsed each database into a list of residues. Third, we rejected residues with incompletely defined structures, as well as inconsistent residues such as nucleotides from DNAmod. Fourth, we normalized the DNA and RNA residues to nucleotide monophosphates and normalized the protein residues to amino acids. For example, we transformed the DNAmod entries to nucleotides by adding deoxyribose monophosphate to each nucleobase. Fifth, we merged the repeated residues. This included residues that had the same molecular structure, that the upstream sources annotated were equivalent, or that had similar names. Lastly, we identified the indices of the forward and reverse bonding sites in each residue. This included the 3' and 5' atoms in each DNA and RNA residue and the carboxyl and amino atoms in each protein residue.

We automated the alphabet construction process by writing scripts to build each alphabet. Going forward, this will enable us to periodically incorporate updates to the upstream databases into the alphabets.

6. Construction of the ontology of crosslinks

We developed the ontology of crosslinks based on entries in RESID which represent crosslinked dimers. First, we searched RESID for entries which represent crosslinked dimers. Second, we identified the individual residues which participate in each dimer. Third, we used ChemAxon Marvin [9] to identify the atoms involved in each crosslink. Next, we used Open Babel [10] to determine the indices of these atoms. Finally, we manually assigned an id and name to each crosslink.

7. Semantic verification of polymers and complexes

To help quality control information about macromolecules, *BpForms* and *BcForms* have methods for verifying the semantic correctness of polymers and complexes. The *BpForms* verification method checks that each residue has a defined structure, each atom that bonds an adjacent residue has a defined element and position which is consistent with the structure of its parent residue, and each pair of consecutive residues can form a bond. The method also checks that the element and position of each atom in each crosslink are consistent with the structure of its parent residue.

The *BcForms* verification method checks that each subunit is semantically concrete and the element and position of each atom in each crosslink are consistent with the structure of its parent residue. For example, these methods can identify invalid proteins that contain consecutive residues which cannot bond because the first residue lacks a carboxyl terminus or the second residue lacks an amino terminus.

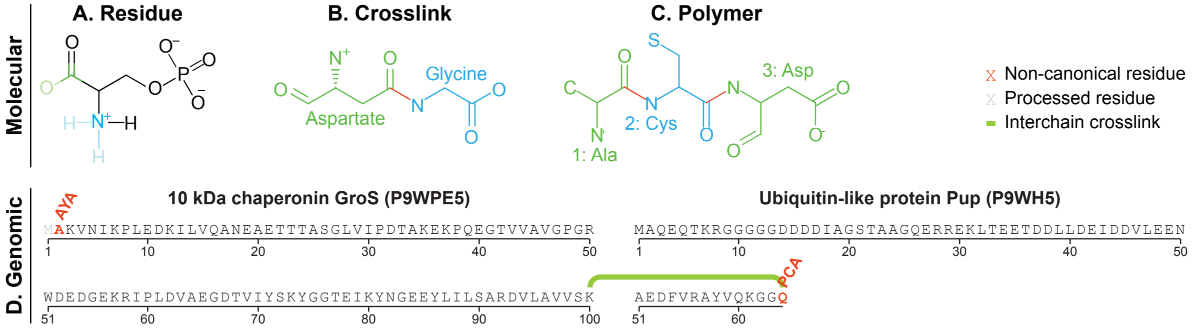


Figure S1. Molecular and genomic visualizations of polymers and complexes that can be generated with *BpForms* and *BcForms*. *BpForms* can generate molecular visualizations of residues such as phosphoserine (**A**), crosslinks such as an isoaspartyl glycine isopeptide bond (**B**), and polymers such as the tripeptide ACD (**C**). The blue and green letters in (A) indicate the atoms which can bond with preceding and following residues; the light blue and light green letters indicate the atoms which are displaced by the formation of these bonds. The blue and green elements in (B) indicate the individual residues involved in the crosslink; the red line indicates the covalent bond that crosslinks the residues. The green elements in (C) indicate the first and third residues in the peptide, the blue elements indicate the second residue, and the red lines indicate the covalent bonds between the residues. *BpForms* and *BcForms* can also generate sequence-based visualizations of polymers and complexes such as the pupylation of chaperonin GroS (**D**, UniProt: P9WPE5, P9WH5). The gray letter indicates a residue which is removed post-translationally, the red letters indicate the residues which are post-translationally modified, and the green line indicates the residues which are post-translationally crosslinked.

8. Visualizations of polymers and complexes

BpForms and *BcForms* can generate several molecular and genomic visualizations of residues, crosslinks, polymers, and complexes. Figure S1 contains examples of these visualizations.

9. Integration with omics and systems and synthetic biology formats

9.1. FASTA format for DNA, RNA, and protein sequences

Box S1 illustrates how *BpForms* can be integrated with the FASTA format [11] to describe multiple non-canonical DNA, RNA, or proteins within a single file. The *BpForms* Python library includes methods for importing and exporting *BpForms* to and from FASTA documents. *BpForms*-encoded FASTA documents can also be read and written by standard FASTA tools such as Biopython [12].

9.2. BioPAX format for pathways

BioPAX [13] is a format for describing biochemical pathways such as metabolism. Box S2 illustrates how *BpForms* can be integrated with BioPAX to describe the polymers that participate in pathways. Users who need to describe residues and crosslinks which are not part of the public *BpForms* ontologies can either describe the residues and crosslinks inline inside descriptions of polymers, build custom alphabets of residues and a custom ontology of crosslinks and bundle these documents with BioPAX documents into COMBINE archives [14], or submit Git pull requests to add residues and crosslinks to the public alphabets and ontology. By helping BioPAX describe the polymers involved in pathways, *BpForms* can make pathways easier to understand and combine into comprehensive maps of cells. Unfortunately, there is no straightforward way to integrate *BcForms* with BioPAX because BioPAX's data model for complexes is not extensible.

9.3. CellML and SBML formats for kinetic models

CellML [15] and the Systems Biology Markup Language (SBML) [16] are formats for describing kinetic models. Both formats have limited capabilities to describe the semantic meaning of model elements which represent macromolecules because both formats encourage users to annotate the meaning of model elements by referencing entities in databases such as UniProt [17] which do not represent every possible form of every macromolecule. For example, CellML and SBML cannot capture the differences among the monophosphorylated states of MAPK because UniProt does not have distinct entries for each state.

Boxes S3 and S4 illustrate how *BpForms* and *BcForms* can be integrated with CellML and SBML to describe the macromolecules represented by models. Users who need to describe residues and crosslinks which are not part of the public *BpForms* ontologies can either describe the residues and crosslinks inline inside descriptions of polymers, build custom alphabets of residues and a custom ontology of crosslinks and bundle these documents with CellML and SBML documents into COMBINE archives [14], or submit Git pull requests to add residues and crosslinks to the public ontologies.

By describing the semantic meaning of model elements, *BpForms* and *BcForms* can make models easier to understand, compare, extend, and combine into more comprehensive models such as whole-cell (WC) models [18, 19].

9.4. Synthetic Biology Open Language (SBOL) format for genetic designs

SBOL [22] is a format for describing genetic designs for synthetic organisms. Box S5 illustrates how *BpForms* can be integrated with SBOL to concretely describe the DNA, RNA, and protein parts of genetic designs. Users who need to describe residues and crosslinks which are not part of the public *BpForms* ontologies can either describe the residues and crosslinks inline inside descriptions of polymers, build custom alphabets of residues and a custom ontology of crosslinks and bundle these documents with SBOL documents into COMBINE archives [14], or submit Git pull requests to add residues and crosslinks to the public alphabets and ontology. In SBOL PEP 033 [23], we formally proposed this integration between *BpForms* and SBOL to the SBOL community. Unfortunately, there is no straightforward way to integrate *BcForms* with SBOL because SBOL's data model for complexes is not extensible. Instead, we encourage the SBOL community to expand SBOL to capture stoichiometric, crosslink, and nick information about complexes.

By helping capture the structures of parts, *BpForms* can help bioengineers identify the biosynthetic dependencies of parts and, in turn, identify constraints on the transformation of parts into new hosts. For example, *BpForms* can help bioengineers identify post-translational modification enzymes that must be co-transformed with parts to synthesize modifications that are essential to the parts.

```
> yp | phosphorylated MEK | Q02750 | pS218
MPKKKPTPIQLNPAPDGSAVNGTSSAETNLEALQKKLEELELDEQQRKRLEAFLTQKQKVGELKDDDFEKISELGAGNGGVVFKV
SHKPSGLVMARKLIHLEIKPAIRNQIIRELQVLHECNSPYIVGFYGAFYSDGEISICMEHMDGGSLDQVLKKAGRIPEQILGKVS
IAVIKGLTYLREKHKIMHRDVKPSNILVNSRGEIKLCDFGVSGQLID{AA0037}MANSFVGTRSYMSPERLQGTHYSVQSDIWS
MGLSLVEMAVGRYPIPPPDAKELELMFGCQVEGDAAETPPRPRTPGRPLSSYGMDSRPPMAIFELLDYIVNEPPPKLPSGVFSLE
FQDFVNKCLIKNPAERADLKQLMVHAFIKRSDAEEVDFAGWLCSTIGLNQPSTPTHAAGV
```

Box S1. *BpForms* can be integrated with the FASTA format to describe multiple polymers within a single document. For example, a FASTA document can contain a *BpForms*-encoded description of monophosphorylated MAPK (UniProt: Q02750).

```
...
<bp:DNA>
   <bp:entityReference>
      <bp:DNAReference>
         <bp:sequence
            rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            rdf:about="http://edamontology.org/format_3909#dna">
            ...
            TGATTTGCCGTGGCGAGAAAATGTCG{a}TCGCCATTATGGCCGGCGTATTAGAAGCGCGCGGTCAC
            AACGTTACTGTTATCG{a}TCCGGTCGAAAAACTGCTGGCAGTGGGGCATTACCTCGAATCTACCGT
            ...
         </bp:sequence>
      </bp:DNAReference>
   </bp:entityReference>
</bp:DNA>
...
```

Box S2. ***BpForms* can help BioPAX documents describe the polymers involved in pathways.** For example, *BpForms* can help BioPAX capture the DNA methylation (orange) that helps *Escherichia coli* detect and degrade foreign DNA. Positions 701 to 800 of *E. coli*'s genome are shown.

10. Implementation

We implemented *BpForms* and *BcForms* with Python 3 [24] and several additional packages. We described the grammar in EBNF, and used Lark [2] to implement a parser for the grammar. We built the alphabets using BeautifulSoup [25], ChemAxon Marvin [9], Open Babel [10], Requests [26], and SQLAlchemy [27]. We described the alphabets in YAML Ain't Markup Language [28], and used ruamel.yaml [29] to parse the alphabets. We used Open Babel and Marvin to implement calculations of properties of macromolecules. We used Marvin to implement the molecular visualizations, and implemented the genomic visualizations using Scalable Vector Graphics (SVG) [30]. We used BioPython [12] to implement methods for importing and exporting *BpForms*-encoded polymers to and from FASTA documents. We implemented the command-line interfaces with Cement [31] and implemented the REST APIs with Flask-RESTPlus [32]. We implemented the web applications using Zurb Foundation [33] and FancyBox [34].

We deployed the web applications on a virtual private server using used Passenger [35].

We used the unittest module [36] to develop over 250 tests to verify *BpForms* and *BcForms*. We used Coverage.py [37] to check that our tests are comprehensive.

We used reStructuredText [38] and Sphinx [39] to generate documentation for *BpForms* and *BcForms*. We used Jupyter [40] to develop interactive tutorials for *BpForms* and *BcForms*.

11. Comparison with other formats and alphabet-like resources

11.1. Comparison of *BpForms* grammar with other formats for polymers

Several text formats, such as ProForma [41], have been developed to represent the structure of DNA, RNA, and proteins. In addition, several container formats, such as BioPAX, have limited abilities to represent the structure of DNA, RNA, and proteins.

As described below and summarized in Table S1, we believe that *BpForms* is a better format for omics, systems biology, and synthetic biology research because it abstracts the complete molecular structure of DNA, RNA, and proteins as collections of residues, crosslinks, and nicks; *BpForms* can

```
...

<species metaid="cdc2k" name="cdc2k-p">
   <annotation>
      <rdf:RDF
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
         <rdf:Description rdf:about="#cdc2k">
            <bpforms:ProteinForm xmlns:bpforms="https://bpforms.org">
               MENYQKVEKIGEG{AA0038}{AA0039}GVVYKARHKLSGRIVAMKKIRLEDESEGVPSTAIREISLLKE
               VNDENNRSNCVRLLDILHAESKLYLVFEFLDMDLKKYMDRISETGATSLDPRLVQKFTYQLVNGVNFCHSR
               RIIHRDLKPQNLLIDKEGNLKLADFGLARSFGVPLRNY{AA0038}HEIVTLWYRAPEVLLGSRHYSTGVD
               IWSVGCIFAEMIRRSPLFPGDSEIDEIFKIFQVLGTPNEEVWPGVTLLQDYKSTFPRWKRMDLHKVVPNGE
               EDAIELLSAMLVYDPAHRISAKRALQQNYLRDFH
            </bpforms:ProteinForm>
         </rdf:Description>
      </rdf:RDF>
   </annotation>
</species>

<species metaid="YP" name="p-cyclin">
   <annotation>
      <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
         <rdf:Description rdf:about="#YP">
            <bpforms:ProteinForm xmlns:bpforms="https://bpforms.org">
               MTTRRLTRQHLLANTLGNNDENHPSNHIARAK{AA0037}{AA0037}LH{AA0037}{AA0037}EN{AA
               0037}LVNGKKATVSSTNVPKKRHALDDV{AA0037}NFHNKEGVPLASKNTNVRHTTASVSTRRALEEKS
               IIPATDDEPA{AA0037}KKRRQPSVFNSSVPSLPQHLSTKSHSVSTHGVDAFHKDQATIPKKLKKDVDER
               VVSKDIPKLHRDSVESPESQDWDDLDAEDWADPLMVSEYVVDIFEYLNELEIETMPSPTYMDRQKELAWKM
               RGILTDWLIEVHSRFRLLPETLFLAVNIIDRFLSLRVCSLNKLQLVGIAALFIASKYEEVMCPSVQNFVYM
               ADGGYDEEEILQAERYILRVLEFNLAYPNPMNFLRRISKADFYDIQTRTVAKYLVEIGLLDHKLLPYPPSQ
               QCAAAMYLAREMLGRGPWNRNLVHYSGYEEYQLISVVKKMINYLQKPVQHEAFFKKYASKKFMKASLFVRD
               WIKKNSIPLGDDADEDYTFHKQKRIQHDMKDEEW
            </bpforms:ProteinForm>
         </rdf:Description>
      </rdf:RDF>
   </annotation>
</species>

<species metaid="pM" name="p-cyclin_cdc2-p">
   <annotation>
      <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
         <rdf:Description rdf:about="#YP">
            <bcforms:BcForm xmlns:bcforms="https://bcforms.org">
               YP + cdc2k
            </bcforms:BcForm>
         </rdf:Description>
      </rdf:RDF>
   </annotation>
</species>

...
```

Box S3. ***BpForms* can help SBML describe the semantic meaning of the macromolecules represented by models.** For example, *BpForms* can help SBML describe that the cdc2k variable of the Tyson cell cycle model [20] represents a tri-phosphorylated form of cyclin dependent kinase 1 (UniProt: P04551).

```
...
<component cmeta:id="ypp" name="ypp">
   <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
      <rdf:Description rdf:about="#ypp">
         <bpforms:ProteinForm xmlns:bpforms="https://bpforms.org">
            MPKKKPTPIQLNPAPDGSAVNGTSSAETNLEALQKKLEELELDEQQRKRLEAFLTQKQKVGELKDDDFEKISEL
            GAGNGGVVFKVSHKPSGLVMARKLIHLEIKPAIRNQIIRELQVLHECNSPYIVGFYGAFYSDGEISICMEHMDG
            GSLDQVLKKAGRIPEQILGKVSIAVIKGLTYLREKHKIMHRDVKPSNILVNSRGEIKLCDFGVSGQLID{AA00
            37}MAN{AA0037}FVGTRSYMSPERLQGTHYSVQSDIWSMGLSLVEMAVGRYPIPPPDAKELELMFGCQVEGD
            AAETPPRPRTPGRPLSSYGMDSRPPMAIFELLDYIVNEPPPKLPSGVFSLEFQDFVNKCLIKNPAERADLKQLM
            VHAFIKRSDAEEVDFAGWLCSTIGLNQPSTPTHAAGV
         </bpforms:ProteinForm>
      </rdf:Description>
   </rdf:RDF>
</component>
...
```

Box S4. *BpForms* can help CellML describe the semantic meaning of components which represent macromolecules. For example, *BpForms* can help CellML describe that the ypp variable in the Wang MAPK cascade model [21] represents a biphosphorylated form of MEK (UniProt: Q0275).

capture several types of missing information about polymers; *BpForms* is both human and machine-readable; *BpForms* is backward compatible with the IUPAC/IUBMB format; and *BpForms* can be integrated into omics, systems biology, and synthetic biology formats such as BioPAX, CellML, SBML, and SBOL.

Consistency in representing DNA, RNA, and proteins

Like the IUPAC/IUBMB format, *BpForms* can represent DNA, RNA, and proteins. In contrast, the MODOMICS nomenclature only represents RNA and the Biological Expression Language (BEL) [47], PRO, and ProForma only represent proteins.

We anticipate that *BpForms*' consistent representation of DNA, RNA, and proteins will facilitate the adoption of *BpForms*, as well as facilitate the integration of information about epigenetic, post-transcriptional, and post-translational modification into comprehensive maps, models, and genetic designs.

Capability to concretely represent the chemical structure of polymers

Like molecular formats such as the International Chemical Identifier (InChI) [42], the Protein Data Bank (PDB) format [48] and the Simplified Molecular-Input Line-Entry System (SMILES) [46], *BpForms* can represent the molecular structure of polymers including non-canonical residues,

```
...

<sbol:Sequence>
   <sbol:elements>
      GGGCCUGUAGCUCAGC{8U}GG{8U}{8U}AGAGCGCACGCCUGAU{62A}AGCGUGAG{7G}UCGAUGG{5U}{9U}C
      GAGUCCAUUCAGGCCCACCA
   </sbol:elements>
   <sbol:encoding rdf:resource="http://edamontology.org/format_3909#rna"/>
</sbol:Sequence>

...
```

Box S5. *BpForms* can help SBOL describe DNA, RNA, and protein parts. For example, *BpForms* can help SBOL describe the post-transcriptional modifications required for *Bacillus subtilis* tRNA$^{\mathrm{Ile}}$ 69 (KEGG: BSU_tRNA_69, SynBioHub: BO_28687).

	Format	DNA	RNA	Proteins	NC residues, alphabet	NC residues, user defined	Crosslinks	Nicks	Concrete semantics	Capture knowledge gaps	Abstracts structures	User-defined alphabets	Human-readable	Machine-readable	Software tools	Backward compatible	Composable
Notation	*BpForms*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	InChI [42]	✓	✓	✓		×	✓	✓	✓					✓	✓		✓
	IUPAC/IUBMB [43]	✓	✓	✓					✓		✓	✓		✓	✓	✓	✓
	MODOMICS nomenclature [44]		✓		✓				×		✓	✓				✓	✓
	PRO proteoform format [45]			✓	✓		×	×		×	✓		✓	×			✓
	ProForma [41]			✓	✓	×				×	✓	✓	×			×	✓
	SMILES [46]	✓	✓	✓		×	✓	✓	✓					✓	✓		✓
Container	BEL [47]			✓	✓						✓		✓	✓	✓		✓
	BioPAX [13]	✓	✓	✓	×		✓				✓	✓		✓	✓		
	Protein Data Bank (PDB) format [48]	✓	✓	✓	✓	×	✓	✓	✓		✓			✓	✓		
	SBOL [22]	✓	✓	✓							✓	✓		✓	✓		

Table S1. Comparison between *BpForms* and other formats for describing polymers. Each ✓ indicates a feature of a format; each × indicates a partially-supported feature of a format.

crosslinks, and nicks. In contrast, BEL, BioPAX, the MODOMICS nomenclature, the PRO format, ProForma, and SBOL do not represent the bonding of non-canonical residues or nicks, and only BioPAX has limited abilities to represent crosslinks.

We anticipate that the concrete chemical semantics of *BpForms* will help researchers compare and integrate information into comprehensive networks of cellular biochemistry.

Capability to represent missing information

Similar to the PRO format, *BpForms* can capture several types of missing information about polymers such as the locations non-canonical residues; the structures, masses, and charges of non-canonical residues; and the locations of crosslinks. In contrast, ProForma can only represent missing knowledge about the structures and masses of residues. BEL, BioPAX, InChI, the MODOMICS nomenclature, the PDB format, SBOL, and SMILES cannot represent missing knowledge.

We believe that the ability to represent missing knowledge makes *BpForms* well-suited for omics, WC modeling, and whole-genome engineering which need to represent both knowledge and gaps in knowledge.

Human readability: abstraction of chemistry

BpForms uses alphabets of residues and an ontology of crosslinks to abstract the structures of polymers. These abstractions make *BpForms*-encoded descriptions of polymers easy to read and write. Furthermore, users can define their own abstractions within descriptions of polymers or define their own alphabet of residues or ontology of crosslinks. This enables *BpForms* to represent newly discovered and synthetic residues. BEL, the IUPAC/IUBMB format, the MODOMICS nomenclature, the PRO format, and ProForma are similarly human-readable.

Although the PDB format uses an alphabet, PDB documents are hard to read and write because the format has limited abilities to abstract residues which do not belong to the alphabet, the format has limited abilities to abstract crosslinks, the format does not abstract nicks, and the format is verbose. Molecular formats such as SMILES are not readable for large molecules such as proteins. BioPAX and SBOL are also difficult to read and write because they are verbose.

Machine-readability: formal grammar

Like BEL, BioPAX, InChI, IUPAC/IUBMB, the PDB format, SMILES, and SBOL, *BpForms* is machine-readable because it has a formal grammar. We have used this grammar to build software tools for parsing, validating, calculating properties, exporting, and composing *BpForms*-encoded descriptions of polymers into computational workflows. In contrast, the MODOMICS nomenclature, the PRO format, and ProForma are not machine-readable because we are not aware of formal grammars or software tools for these formats.

Backward compatibility with IUPAC/IUBMB and sequence informatics tools

Like the MODOMICS nomenclature and ProForma, *BpForms* maximizes compatibility with sequence informatics tools by generalizing the IUPAC/IUBMB format. As a result, *BpForms* can be integrated into FASTA documents. In contrast, BEL, BioPAX, InChI, the PDB format, the PRO format, SBOL, and SMILES are less compatible with sequence informatics tools because they are not backward compatible with the IUPAC/IUBMB format.

Composability with other formats

BpForms is compact like other text formats such as BEL, the IUPAC/IUBMB format, the MODOMICS nomenclature, the PRO format, and ProForma. This makes *BpForms* composable with formats for describing entire pathways, models, and genetic design such as BioPAX, CellML, SBML, and SBOL. In contrast, BioPAX, the PDB format, and SBOL are less suited to integration into other formats because they are verbose.

11.2. Comparison of *BpForms* alphabets with other alphabet-like resources

Several databases have been developed to help exchange information about non-canonical DNA, RNA, and proteins residues. As described below and summarized in Table S2, we believe that the *BpForms* alphabets are better suited to omics, systems biology, and synthetic biology research because they represent DNA, RNA, and proteins; they represent concrete chemical structures and bonding sites; and they are the most comprehensive collections of residues.

Consistency in representing DNA, RNA, and proteins

Like the Protein Data Bank (PDB) Chemical Component Dictionary (CCD) [3], the *BpForms* alphabets represent DNA, RNA, and protein residues. This consistency makes *BpForms* easy to use and facilitates the integration of information, models, and genetic designs that involve DNA, RNA, and proteins. In contrast, DNAmod and REPAIRtoire only represent DNA residues, MODOMICS and the RNA Modification Database only represent RNA residues, and the Protein Modification Ontology (MOD) and RESID only represent protein residues.

Concreteness of chemical semantics

Like the PDB CCD, the *BpForms* alphabets define complete residues and each residue defines a concrete chemical structure and concrete bonding sites with the preceding and following residues. This enables *BpForms* to represent the primary structures of non-canonical polymers. This also enables *BpForms* to capture modifications to the sugar-phosphate backbone of DNA and RNA.

Alphabet	**DNA**	**RNA**	**Protein**	**Complete residues**	**Structures**	**Bonding**	**Structure-based**	**Biochemistry-based**
BpForms	422	378	1,435	✓	✓	✓	✓	✓
DNAmod [4] (verified nucleobases)	58				✓			✓
REPAIRtoire [5] (monophosphates)	34			✓	✓			✓
MODOMICS [44]		172			✓			✓
RNA Modification Database [7]		112			✓			✓
PDB CCD [3] (unambiguous released residues)	373	271	1,095	✓	✓	✓	✓	
Protein Modification Ontology (MOD) [49] (leaves)			1,445	×	×		×	✓
RESID [8]			621		✓			✓

Table S2. Comparison between *BpForms* and other collections of DNA, RNA, and protein residues. Each ✓ indicates a feature of a collection; each × indicates a partially-supported feature of a collection.

In contrast, DNAmod, MODOMICS, and RNA Modification Database have limited abilities to represent non-canonical DNA and RNA because these formats do represent the sugar-phosphate backbone and the formats have ambiguous chemical semantics because they do not capture bonding sites; REPAIRtoire and RESID have ambiguous chemical semantics because they do not represent bonding sites; and most MOD entries have ambiguous chemical semantics because they do not define concrete structures or bonding sites.

Breadth of residues from structural and biochemical studies

To make *BpForms* useful for structural biology, omics, systems biology, and synthetic biology, we populated the *BpForms* alphabets with residues that are important for a wide range of research. As a result, the *BpForms* alphabets are the most comprehensive collections of residues. In contrast, the PDB CCD represents fewer residues because it is primarily based on structural biology data and DNAmod, MOD, MODOMICS, REPAIRtoire, the RNA Modification Database, and RESID represent fewer residues because they are mainly based on biochemical data.

11.3. Comparison of the *BpForms* crosslink ontology with other resources

Several resources include information about crosslinks. As illustrated in Table S3, we believe that the *BpForms* crosslink ontology is better suited for omics and systems and synthetic biology research because it concretely represents crosslinks and it is composable with the residues in the *BpForms* alphabets into descriptions of macromolecules. In contrast, REPAIRtoire [5], the Protein Modification Ontology (MOD) [49], and RESID [8] use residues to indirectly represent crosslinks, and the crosslinks in the UniProt controlled vocabulary of posttranslational modifications [17] do not have concrete chemical semantics. As a result, the crosslinks represented by these resources are difficult to compose into macromolecules.

11.4. Comparison of *BcForms* grammar with other formats for complexes

Despite the importance of complexes, only a few formats have been developed to represent complexes. As described below and summarized in Table S4, we believe that *BcForms* is better suited

Resource	Direct representation	Concrete semantics	Composable
BpForms	✓	✓	✓
Protein Data Bank (PDB) [48]	✕	✓	
Protein Modification Ontology (MOD) [49]		✕	
REPAIRtoire [5]		✕	
RESID [8]		✓	
UniProt [17]	✓		

Table S3. Comparison between the *BpForms* crosslinks ontology and other resources that describe crosslinks. Each ✓ indicates a feature of a resource; each ✕ indicates a partially-supported feature of a resource.

for omics, systems biology, and synthetic biology research because it is the first format that abstractly represents the primary structure of complexes and it is human-readable, machine-readable, and composable with formats for network research such as CellML and SBML.

Capability to represent the chemical structure of complexes

Like InChI, the PDB format, and SMILES, *BcForms* can represent the primary structure of complexes, including non-canonical subunits and interchain crosslinks. In contrast, BioPAX and SBOL have limited abilities to represent crosslinks. We anticipate that the concrete semantics of *BcForms* will facilitate the integration of data, models, and genetic designs that involve complexes.

Abstraction, human readability, and composability

Similar to BioPAX, and SBOL, *BcForms* abstracts complexes as sets of subunits and crosslinks. This makes *BcForms* human-readable and composable with formats for systems biology and synthetic biology research such as CellML, SBML, and SBOL. By comparison, InChI and SMILES are less human-readable, and the PDB format is both less human-readable and less composable.

12. Case studies

Table S5 provides additional information for the synthetic biology case study discussed in the main text.

13. Acronyms

APT Advanced Package Tool [50]

BEL Biological Expression Language [47]

BNF Backus-Naur Form

COMBINE Computational Modeling in Biology Network [51]

EBNF Extended Backus-Naur Form [1]

InChI International Chemical Identifier [42]

MOD Protein Modification Ontology [49]

Format	Concrete semantics	Abstracts structures	Human-readable	Machine-readable	Software tools	Composable
BcForms	✓	✓	✓	✓	✓	✓
BioPAX [13]		✓		✓	✓	
International Chemical Identifier (InChI) [42]	✓			✓	✓	✓
Protein Data Bank (PDB) format [48]	✓	×		✓	✓	
Synthetic Biology Open Language (SBOL) [22]		✓		✓	✓	
Simplified Molecular-Input Line-Entry System (SMILES) [46]	✓			✓	✓	✓

Table S4. Comparison between *BcForms* and other formats for describing complexes. Each ✓ indicates a feature of a format; each × indicates a partially-supported feature of a format.

PDB CCD id	RESID id	Name	PDB entries	Proteins	Absence from *E. coli*
HYP	AA0030	4-hydroxyproline	239	Collagen and plant walls	[52–54]
HIC	AA0317	4-methyl-histidine	121	Actin and myosin	[55, 56]
MEN	AA0070	N-methyl asparagine	57	Antennae of photosystem II	[57–59]
FVA		N-formyl-L-valine	23	Gramicidin	
6V1			10		
TQQ			8	Aromatic amine dehydrogenase	
TRX		6-hydroxytryptophan	6	RNA polymerase II	
PSW		3-(sulfanylselanyl)-L-alanine	5		

Table S5. The most common protein residues within the Protein Data Bank (PDB) which cannot be synthesized by *Escherichia coli.* We identified these residues by using *BpForms* to analyze the PDB, and we confirmed the absence of the most frequent residues from *E. coli* via the literature. This information could be used to constrain the design of novel strains of *E. coli.* For example, genetic designs based on *E. coli* should not include photosystem II, or such designs should also include phycobiliprotein asparagine methyltransferase CpcM [58].

PDB Protein Data Bank [48]

PDB CCD PDB Chemical Component Dictionary [3]

PRO Protein Ontology [45]

REST Representational State Transfer

SBML Systems Biology Markup Language [16]

SBOL Synthetic Biology Open Language [22]

SMILES Simplified Molecular-Input Line-Entry System [46]

SVG Scalable Vector Graphics [30]

WC Whole-cell [18, 19]

References

1. Wikipedia. *Extended Backus-Naur Form.* https://en.wikipedia.org/wiki/Extended_Backus-Naur_form (2019).

2. Shinan, E. *Lark – a modern parsing library for Python.* https://lark-parser.readthedocs.io.

3. Westbrook, J. D., Shao, C., Feng, Z., Zhuravleva, M., Velankar, S. & Young, J. The Chemical Component Dictionary: complete descriptions of constituent molecules in experimentally determined 3D macromolecules in the Protein Data Bank. *Bioinformatics* **31,** 1274–1278 (2014).

4. Sood, A. J., Viner, C. & Hoffman, M. M. DNAmod: the DNA modification database. *J. Cheminform.* **11,** 30 (2019).

5. Milanowska, K., Krwawicz, J., Papaj, G., Kosiński, J., Poleszak, K., Lesiak, J., Osińska, E., Rother, K. & Bujnicki, J. M. REPAIRtoire–a database of DNA repair pathways. *Nucleic Acids Res.* **39,** D788–D792 (2010).

6. Machnicka, M. A., Milanowska, K., Osman Oglou, O., Purta, E., Kurkowska, M., Olchowik, A., Januszewski, W., Kalinowski, S., Dunin-Horkawicz, S., Rother, K. M., *et al.* MODOMICS: a database of RNA modification pathways–2013 update. *Nucleic Acids Res.* **41,** D262–D267 (2012).

7. Cantara, W. A., Crain, P. F., Rozenski, J., McCloskey, J. A., Harris, K. A., Zhang, X., Vendeix, F. A., Fabris, D. & Agris, P. F. The RNA Modification Database, RNAMDB: 2011 update. *Nucleic Acids Res.* **39,** D195–D201 (2010).

8. Garavelli, J. S. The RESID Database of Protein Modifications as a resource and annotation tool. *Proteomics* **4,** 1527–1533 (2004).

9. ChemAxon Limited. *Marvin.* https://chemaxon.com/products/marvin (2019).

10. O'Boyle, N. M., Guha, R., Willighagen, E. L., Adams, S. E., Alvarsson, J., Bradley, J.-C., Filippov, I. V., Hanson, R. M., Hanwell, M. D., Hutchison, G. R., *et al.* Open data, open source and open standards in chemistry: the Blue Obelisk five years on. *J. Cheminform.* **3,** 37 (2011).

11. Pearson, W. R. Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol.* **183,** 63–98 (1990).

12. Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., *et al.* Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25,** 1422–1423 (2009).

13. Demir, E., Cary, M. P., Paley, S., Fukuda, K., Lemer, C., Vastrik, I., Wu, G., D'eustachio, P., Schaefer, C., Luciano, J., *et al.* The BioPAX community standard for pathway data sharing. *Nat. Biotechnol.* **28,** 935 (2010).

14. Bergmann, F. T., Rodriguez, N. & Le Novère, N. COMBINE archive specification version 1. *J. Integr. Bioinform.* **12,** 104–118 (2015).

15. Cuellar, A., Hedley, W., Nelson, M., Lloyd, C., Halstead, M., Bullivant, D., Nickerson, D., Hunter, P. & Nielsen, P. The CellML 1.1 specification. *J. Integr. Bioinform.* **12,** 4–85 (2015).

16. Hucka, M., Bergmann, F. T., Dräger, A., Hoops, S., Keating, S. M., Le Novère, N., Myers, C. J., Olivier, B. G., Sahle, S., Schaff, J. C., *et al.* The Systems Biology Markup Language (SBML): language specification for level 3 version 2 core. *J. Integr. Bioinform.* **15** (2018).

17. UniProt Consortium *et al.* UniProt: the universal protein knowledgebase. *Nucleic Acids Res.* **46,** 2699 (2018).

18. Karr, J. R., Sanghvi, J. C., Macklin, D. N., Gutschow, M. V., Jacobs, J. M., Bolival Jr, B., Assad-Garcia, N., Glass, J. I. & Covert, M. W. A whole-cell computational model predicts phenotype from genotype. *Cell* **150,** 389–401 (2012).

19. Goldberg, A. P., Szigeti, B., Chew, Y. H., Sekar, J. A., Roth, Y. D. & Karr, J. R. Emerging whole-cell modeling principles and methods. *Curr. Opin. Biotechnol.* **51,** 97–102 (2018).

20. Tyson, J. J. Modeling the cell division cycle: cdc2 and cyclin interactions. *Proc. Natl. Acad. Sci. U. S. A.* **88,** 7328–7332 (1991).

21. Wang, C.-C., Cirit, M. & Haugh, J. M. PI3K-dependent cross-talk interactions converge with Ras as quantifiable inputs integrated by Erk. *Mol. Syst. Biol.* **5** (2009).

22. Cox, R. S., Madsen, C., McLaughlin, J. A., Nguyen, T., Roehner, N., Bartley, B., Beal, J., Bissell, M., Choi, K., Clancy, K., *et al.* Synthetic Biology Open Language (SBOL) version 2.2.0. *J. Integr. Bioinform.* **15** (2018).

23. Karr, J. R. *SBOL SEP 033 – Concrete descriptions of non-canonical DNA, RNA, and proteins.* https://github.com/SynBioDex/SEPs/blob/master/sep_033.md.

24. Python Software Foundation. *Python.* https://python.org/ (2019).

25. Richardson, L. *Beautiful Soup.* https://www.crummy.com/software/BeautifulSoup/ (2019).

26. Reitz, K., Cordasco, I. & Prewit, N. *Requests: HTTP for humans.* https://2.python-requests.org (2019).

27. Bayer, M. *SQLAlchemy – The database toolkit for Python.* https://www.sqlalchemy.org/ (2019).

28. Ben-Kiki, O., Evans, C. & döt Net, I. *YAML: YAML Ain't Markup Language.* https://yaml.org (2019).

29. Van der Neut, A. *ruamel.yaml.* https://yaml.readthedocs.io (2019).

30. W3C SVG Working Group. *Scalable Vector Graphics (SVG).* https://www.w3.org/Graphics/SVG/ (2019).

31. Data Folk Labs. *Cement framework.* https://builtoncement.com/ (2019).

32. Haustant, A. *Flask-RESTPlus.* https://flask-restplus.readthedocs.io (2019).

33. ZURB, Inc. *Foundation – The most advanced responsive front-end framework in the world.* https://foundation.zurb.com (2019).

34. Skarnelis, J. *FancyBox – Fancy jQuery lightbox alternative.* http://fancybox.net (2019).

35. Phusion Holding B.V. *Passenger.* https://www.phusionpassenger.com (2019).

36. Python Software Foundation. *unittest unit testing framework.* https://docs.python.org/3/library/unittest.html (2019).

37. Batchelder, N. *Coverage.py.* https://coverage.readthedocs.io (2019).

38. Goodger, D. *reStructuredText.* http://docutils.sourceforge.net/rst.html (2019).

39. Brandl, G. *et al. Sphinx – Python documentation generator.* http://www.sphinx-doc.org (2019).

40. Project Jupyter. *Jupyter.* https://jupyter.org (2019).

41. LeDuc, R. D., Schwämmle, V., Shortreed, M. R., Cesnik, A. J., Solntsev, S. K., Shaw, J. B., Martin, M. J., Vizcaino, J. A., Alpi, E., Danis, P., *et al.* ProForma: a standard proteoform notation. *J. Proteome Res.* **17,** 1321–1325 (2018).

42. Heller, S. R., McNaught, A., Pletnev, I., Stein, S. & Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. *J. Cheminform.* **7,** 23 (2015).

43. Leonard, S. A. IUPAC/IUB single-letter codes within nucleic acid and amino acid sequences. *Curr. Protoc. Bioinformatics,* A–1A (2003).

44. Boccaletto, P., Machnicka, M. A., Purta, E., Piątkowski, P., Bagiński, B., Wirecki, T. K., de Crécy-Lagard, V., Ross, R., Limbach, P. A., Kotter, A., *et al.* MODOMICS: a database of RNA modification pathways. 2017 update. *Nucleic Acids Res.* **46,** D303–D307 (2017).

45. Natale, D. A., Arighi, C. N., Blake, J. A., Bona, J., Chen, C., Chen, S.-C., Christie, K. R., Cowart, J., D'Eustachio, P., Diehl, A. D., *et al.* Protein Ontology (PRO): enhancing and scaling up the representation of protein entities. *Nucleic Acids Res.* **45,** D339–D346 (2016).

46. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inform. Comp. Sci.* **28,** 31–36 (1988).

47. Fluck, J., Madan, S., Ansari, S., Karki, R., Rastegar-Mojarad, M., Catlett, N. L., Hayes, W., Szostak, J., Hoeng, J., Peitsch, M., *et al.* Training and evaluation corpora for the extraction of causal relationships encoded in biological expression language (BEL). *Database* **2016** (2016).

48. Westbrook, J. D. & Fitzgerald, P. in *Structural Bioinformatics* (eds Bourne, P. E. & Weissig, H.) 161–179 (Wiley Online Library, 2003).

49. Montecchi-Palazzi, L., Beavis, R., Binz, P.-A., Chalkley, R. J., Cottrell, J., Creasy, D., Shofstahl, J., Seymour, S. L. & Garavelli, J. S. The PSI-MOD community standard for representation of protein modification data. *Nat. Biotechnol.* **26,** 864 (2008).

50. Ubuntu Documentation Team. *Apt* https://help.ubuntu.com/lts/serverguide/apt.html.

51. Hucka, M., Nickerson, D. P., Bader, G. D., Bergmann, F. T., Cooper, J., Demir, E., Garny, A., Golebiewski, M., Myers, C. J., Schreiber, F., *et al.* Promoting coordinated development of community-based information standards for modeling in biology: the COMBINE initiative. *Front. Bioeng. Biotechnol.* **3,** 19 (2015).

52. Pinkas, D. M., Ding, S., Raines, R. T. & Barron, A. E. Tunable, post-translational hydroxylation of collagen domains in Escherichia coli. *ACS Chem. Biol.* **6,** 320–324 (2011).

53. An, B., Kaplan, D. L. & Brodsky, B. Engineered recombinant bacterial collagen as an alternative collagen-based biomaterial for tissue engineering. *Front. Chem.* **2,** 40 (2014).

54. Yi, Y., Sheng, H., Li, Z. & Ye, Q. Biosynthesis of trans-4-hydroxyproline by recombinant strains of Corynebacterium glutamicum and Escherichia coli. *BMC Biotechnol.* **14,** 44 (2014).

55. Kwiatkowski, S., Seliga, A. K., Vertommen, D., Terreri, M., Ishikawa, T., Grabowska, I., Tiebe, M., Teleman, A. A., Jagielski, A. K., Veiga-da Cunha, M., *et al.* SETD3 protein is the actin-specific histidine N-methyltransferase. *Elife* **7,** e37921 (2018).

56. Xiao, H., Peters, F. B., Yang, P.-Y., Reed, S., Chittuluru, J. R. & Schultz, P. G. Genetic incorporation of histidine derivatives using an engineered pyrrolysyl-tRNA synthetase. *ACS Chem. Biol.* **9,** 1092–1096 (2014).

57. Klotz, A. & Glazer, A. N. gamma-N-methylasparagine in phycobiliproteins. Occurrence, location, and biosynthesis. *J. Biol. Chem.* **262,** 17350–17355 (1987).

58. Shen, G., Leonard, H. S., Schluchter, W. M. & Bryant, D. A. CpcM posttranslationally methylates asparagine-71/72 of phycobiliprotein beta subunits in Synechococcus sp. strain PCC 7002 and Synechocystis sp. strain PCC 6803. *J. Bacteriol.* **190,** 4808–4817 (2008).

59. Scheer, H & Zhao, K.-H. Biliprotein maturation: the chromophore attachment. *Mol. Microbiol.* **68,** 263–276 (2008).