K

Kappa, κ

► κ-Calculus

Kauffman Network

► Boolean Networks

κ-Calculus

Alida Palmisano[1] and Corrado Priami[2]
[1]Department of Biological Sciences and Department of Computer Science, Department of Biological Sciences Virginia Tech, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
[2]Microsoft Research-University of Trento Centre for Computational and Systems Biology and DISI, University of Trento, Povo, Trento, Italy

Synonyms

Kappa, κ

Definition

κ is as a rule-based language for modeling protein interaction networks that allows the formalization of molecular agents and their interactions in signaling networks (Danos et al. 2007, 2008).

The κ description of a system consists of a collection of agents and rules. An agent has a name and a number of labeled sites, collectively referred to as the agent's interface. A site may have an internal state, typically used to denote its phosphorylation status or other post-translational modifications. Rules provide a concise description of how agents interact. Elementary interactions consist of the binding or unbinding of two agents, the modification of the state of a site, and the deletion or creation of an agent.

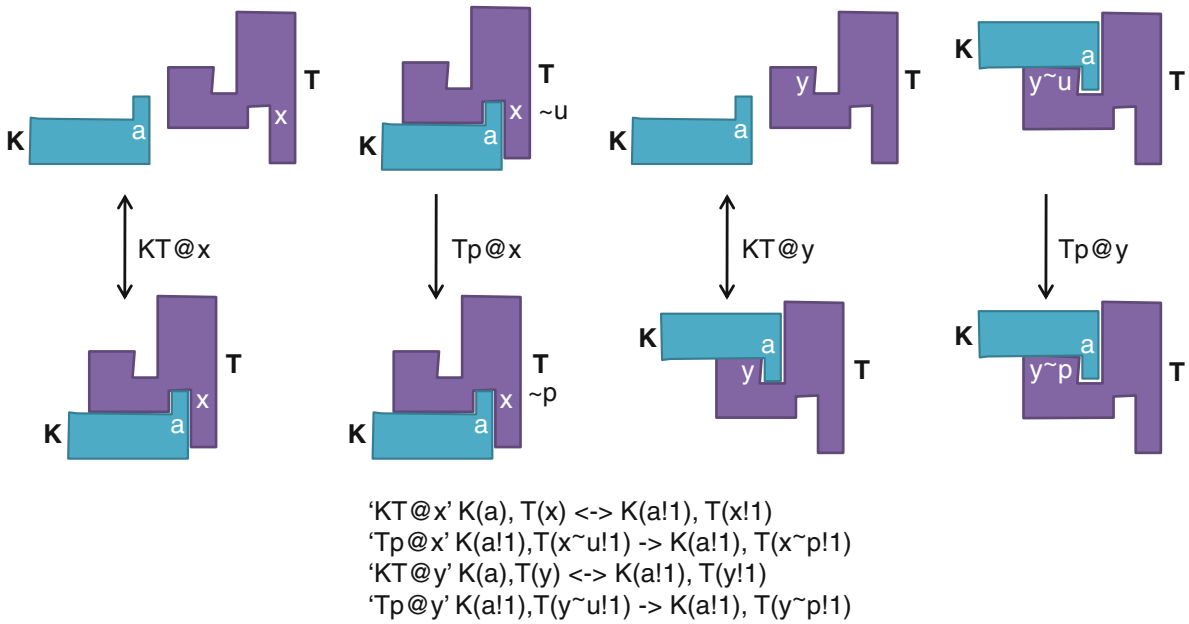
The rule-based modeling approach of κ incorporates causality constraints in the rules by using partial complexes: Only the aspects of the state of a complex which matter for an event to happen need to be specified. This reliance on partial complexes allows to capture compact descriptions and work around the huge numbers of combinations one would have to contemplate (or neglect) otherwise. The more detailed the partial complex, that is to say the less partial, the more conditions must be met for a particular event to happen.

It is possible to associate rate constants with each rule and the rule has to be expressed in their elementary form.

The language is equipped with a visual notation, where proteins are represented by boxes with domains on their boundaries. The calculus is provided with an exact stochastic simulator, and a series of tools (that can be found at http://www.kappalanguage.org/) that allow different kinds of analyses on κ-calculus models.

In Fig. 1, we report part of small example (taken from Danos et al. 2007) just to show the basic primitives of the language.

W. Dubitzky et al. (eds.), *Encyclopedia of Systems Biology*, DOI 10.1007/978-1-4419-9863-7,



κ-Calculus, Fig. 1 Consider two agents: a kinase K and a target T with two phosphorylatable sites x and y. The behavior that needs to be modeled is the set of the following three elementary actions (i.e., rules) (1) the kinase K binds its target T either at site x or y; (2) the kinase may (but need not) phosphorylate the site to which it is bound; (3) the kinase dissociates (unbinds) from its target. The code presented in the lower part of the figure models the interaction between a kinase K and its target T. The rules are labeled with a mnemonic on the left. "~u" (unphosphorylated) and "~p" (phosphorylated) represent the internal state of an interface, and physical associations (bindings or links) are indicated by "!" with shared indices across agents to indicate the two endpoints of a link. The left hand side of a rule specifies a condition in the form of a pattern expressed as a partial graph, which represents binding states and site values of agents. The right hand side of a rule specifies (usually elementary) changes to agents mentioned on the left. A *double arrow* indicates a reversible rule, the name refers to the forward version of the rule say r, while the opposite rule is written r_op. KT@x and KT@y are the rules for the binding of K and T on site x and y respectively. Tp@x and Tp@y are the rules for the possible phosphorylation action of K on T. The dissociation of the two agents is encoded in the reversibility of the first two rules

Cross-References

▶ Cell Cycle Modeling, Process Algebra

References

Danos V, Feret J, Fontana W, Harmer R, Krivine J (2007) Rule-based modelling of cellular signalling. In: CONCUR. Lecture Notes in Computer Science, vol 4703. Springer, pp 17–41

Danos V, Feret J, Fontana W, Harmer R, Krivine J (2008) Rule-based modelling, symmetries, refinements. In: Proceedings of FMSB'08. Lecture Notes in Computer Science, vol 5054. Springer, pp 103–122

k-Cone Space

▶ Dynamic Metabolic Networks, k-Cone

KEGG PATHWAY

▶ KEGG Pathway Database

KEGG Pathway Database

Yu-Qing Qiu
Department of Chemical Pathology, The Chinese University of Hong Kong, Hong Kong, China

Synonyms

KEGG PATHWAY; Kyoto Encyclopedia of Genes and Genomes Pathway

Definition

KEGG (Kanehisa et al. 2010) (Kyoto Encyclopedia of Genes and Genomes) pathway database is a well-known publicly accessible pathway database. It is one main database of KEGG, which was built in 1995, and is a bioinformatics resource as part of the research projects of the Kanehisa Laboratories in the Bioinformatics Center of Kyoto University and the Human Genome Center of the University of Tokyo. KEGG PATHWAY contains our knowledge curated from scientific literatures on the biological molecular interaction and reaction networks, including protein-protein interaction, protein-DNA binding, protein-ligand interaction, enzyme-mediated biomass reaction, etc. Interactions within one specific biological process or function are drawn manually to pathway maps. By far, there are 365 pathway maps collected from 113,760 references, which are categorized into metabolism, genetic information processing, environmental information processing, cellular processes, organismal systems, human diseases, and drug development.

References

Kanehisa M, Goto S, Furumichi M, Tanabe M, Hirakawa M. KEGG for representation and analysis of molecular networks involving diseases and drugs. Nucleic Acids Res. 2010;38: D355–D360.

Kernel Machines

▶ Learning, Kernel-based

Kernel Methods

▶ Learning, Kernel-based

Key Step

▶ Rate-limiting step

Killer Cells

▶ Natural Killer Cells, Mycobacterial Infection

Kinase Activity Assay

▶ Protein Kinase Assay

Kinase Assay

▶ Protein Kinase Assay

Kinetic Equations

▶ Cell Cycle Modeling, Differential Equation

Kinetic Modeling and Simulation

Sang Yup Lee
Department of Chemical and Biomolecular Engineering and Department of Bio and Brain Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea

Synonyms

Dynamic modeling and simulation

Definition

In systems where (bio)chemical reactions take place, kinetic modeling and simulation refer to mathematical description of changes in properties of the system of interest, for instance, concentrations of metabolites, proteins, or other cellular components, and reaction fluxes in the case of biological system with respect to time. Dynamic properties of biological system often start from dynamic mass balance of cellular components of interest and can be implicitly described as

$$\frac{dx}{dt} = S \cdot v \quad (1)$$

where S is the stoichiometric matrix of cellular components involved in biochemical reactions, v is the vector of reaction rates, and x is the vector of concentrations of the considered cellular components.

References

Fogler HS (2005) Elements of chemical reaction engineering. Prentice Hall, Upper Saddle River

Kinetic Modeling of microRNA Regulation

▶ Computational microRNA Biology

Kinetic Parameter Information Resource, KiPar

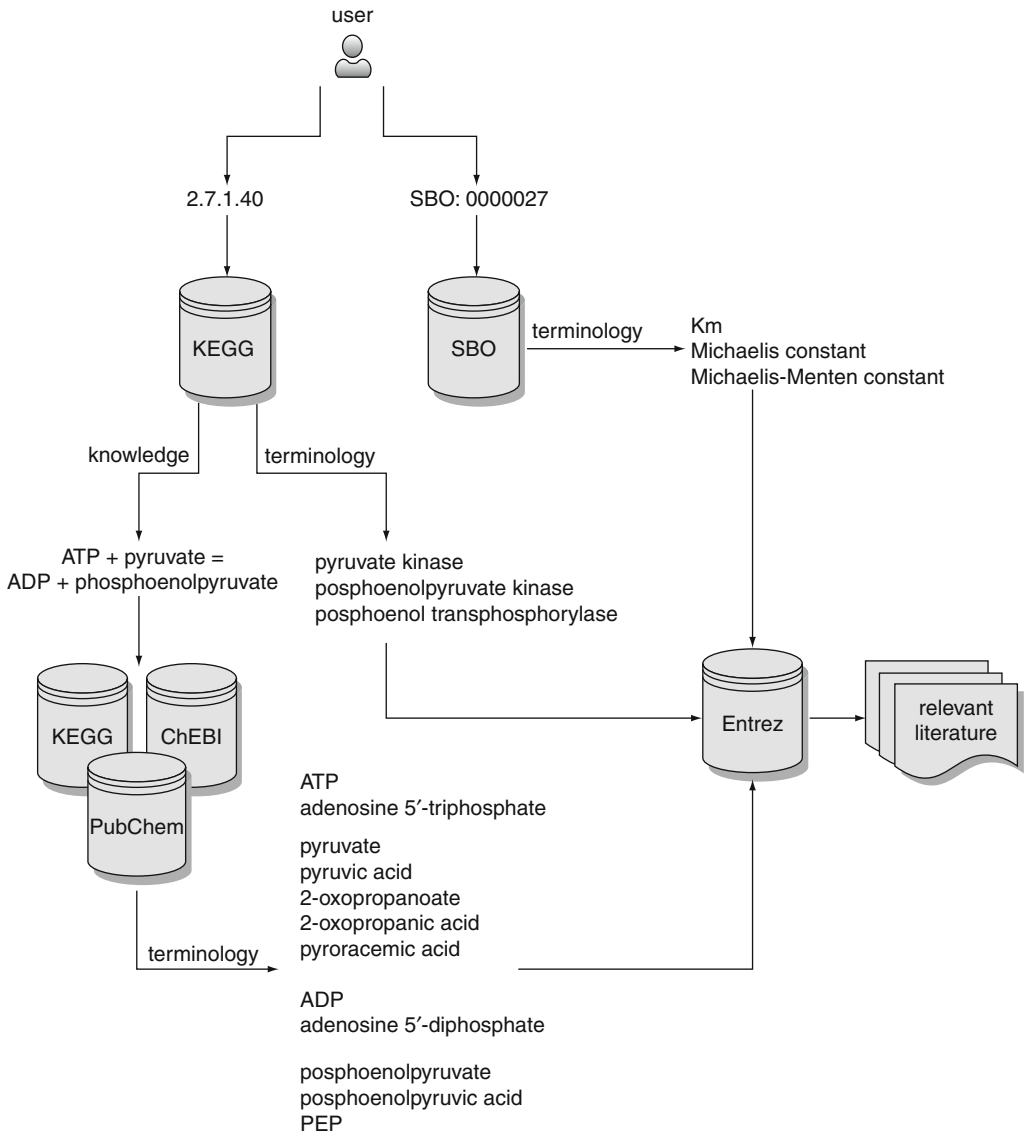
Irena Spasić[1] and Douglas Bruce Kell[2]
[1]School of Computer Science and Informatics, Cardiff University, Cardiff, UK
[2]School of Chemistry and Manchester Interdisciplinary Biocentre, University of Manchester, Manchester, UK

Definition

KiPar is an ▶ information retrieval system designed to facilitate access to the literature relevant for kinetic modeling of metabolic pathways in yeast. Information supplied as user input includes the enzymes catalyzing the reactions of interest and the parameters whose values are required for kinetic modeling. The output is produced as a list of documents (either abstracts from ▶ PubMed or full-text articles from PubMed Central) that should contain the required values of kinetic parameters. There are two groups of users of this specific application: (1) experimentalists who wish to compare experimentally estimated values of kinetic parameters to those reported in the literature, and (2) mathematical modelers who wish to incorporate known values of kinetic parameters into metabolic models.

Characteristics

A typical systems biology network modeling strategy has four main parts (Kell 2006; Herrgård et al. 2008): two qualitative ones, in which the reaction partners and their modifiers are set down, and two more quantitative ones, in which the kinetic rate equations for each step are described, together with the values of their parameters (such as K_m and k_{cat}). The latter two in particular provide researchers with considerable challenges, as the necessary data are often buried deep inside individual papers (Ananiadou et al. 2006; Hakenberg et al. 2004). KiPar provides an integrative approach, combining a number of publicly available data and software resources, for effective retrieval of documents relevant for the kinetic modeling of metabolic pathways (Spasić et al. 2009), which has been recognized as one of the principal goals of systems biology (Palsson 2006). The input information is provided by the user as a set of identifiers, which include EC numbers to specify enzymes (e.g., EC 2.7.1.1 for *hexokinase*) and Systems Biology Ontology (SBO) terms to specify kinetic parameters (e.g., SBO:0000025 for k_{cat}). These identifiers are used as entry points into the relevant public resources of biological knowledge: ▶ KEGG ENZYME (Kanehisa et al. 2008) and Systems Biology Ontology (Le Novère 2006). The names of the identified entities, including synonyms, are collected automatically from these resources to be used later as search terms against the literature databases: ▶ PubMed and PubMed Central. In addition, ▶ KEGG ENZYME is queried for other types of information related to the given enzymes, such as the compounds that participate in the corresponding reactions. As before, the names of the identified compounds are collected from the cross-referenced databases: KEGG COMPOUND, PubChem (Bolton et al. 2008), and ChEBI (Degtyarenko et al. 2008). The gathered synonyms referring to the enzyme and the compounds involved in a reaction are combined together to search the literature for information on the given reaction, which is usually not designated by a name that could be used as a search term (Ananiadou et al. 2010). The search results for individual reactions are further



Kinetic Parameter Information Resource, KiPar, Fig. 1 Simplified structure of the system, given as a logical sequence of the elementary acts performed. Given the input specification about an enzyme and its kinetics, KiPar collects the associated terminology (i.e., enzyme names as well as the names of compounds acting as substrates/products in the corresponding reaction, and the terms referring to the given kinetic parameter) from publicly available biological databases. The terms collected are used to search the literature for enzyme kinetic parameters

filtered out using the search terms gathered for the kinetic parameters. Finally, the content of the retrieved documents is annotated with the matching search terms in order to allow the user easier identification of the relevant information. Figure 1 illustrates a simplified version of the information flow in KiPar.

KiPar makes extensive use of domain knowledge, most of which is accessed dynamically through the web services of the selected sources. Some of the relevant domain knowledge is incorporated into the formula for scoring the document relevance. These facts free the user from formulating complex search queries involving the use of knowledge and terminology concerned with the relevant entities and their relations, which otherwise make manual searching for enzyme kinetic parameters complex and time-consuming. This approach has been found to perform better than the traditional Boolean search.

In addition, multiple reactions and their kinetic parameters can be specified in a single search request, rather than one reaction at a time, which further facilitates access to the literature discussing enzyme kinetic parameters required for developing large-scale metabolic models as is the case in systems biology.

Cross-References

- ▶ Information Retrieval
- ▶ KEGG PATHWAY
- ▶ MEDLINE and PubMed

References

Ananiadou S, Kell DB, Tsujii J (2006) Text mining and its potential applications in systems biology. Trends Biotechnol 24:571–579

Ananiadou S, Pyysalo S, Tsujii J, Kell DB (2010) Event extraction for systems biology by text mining the literature. Trends Biotechnol 28:381–390

Bolton EE, Wang Y, Thiessen PA, Bryant SH (2008) PubChem: integrated platform of small molecules and biological activities. Annu Rep Comput Chem 4:217–241

Degtyarenko K, Matos PD, Ennis M et al (2008) ChEBI: a database and ontology for chemical entities of biological interest. Nucleic Acids Res 36:D344–D350

Hakenberg J, Schmeier S, Kowald A, Klipp E, Leser U (2004) Finding kinetic parameters using text mining. Omics 8:131–152

Herrgård MJ, Swainston N, Dobson P, Dunn WB, Arga KY, Arvas M, Blüthgen N, Borger S, Costenoble R, Heinemann M, Hucka M, Le Novère N, Li P, Liebermeister W, Mo ML, Oliveira AP, Petranovic D, Pettifer S, Simeonidis E, Smallbone K, Spasić I, Weichart D, Brent R, Broomhead DS, Westerhoff HV, Kırdar B, Penttilä M, Klipp E, Palsson BØ, Sauer U, Oliver SG, Mendes P, Nielsen J, Kell DB (2008) A consensus yeast metabolic network obtained from a community approach to systems biology. Nat Biotechnol 26:1155–1160

Kanehisa M, Araki M, Goto S, Hattori M, Hirakawa M, Itoh M, Katayama T, Kawashima S, Okuda S, Tokimatsu T, Yamanishi Y (2008) KEGG for linking genomes to life and the environment. Nucleic Acids Res 36:D480–D484

Kell DB (2006) Metabolomics, modelling and machine learning in systems biology: towards an understanding of the languages of cells. The 2005 Theodor Bücher lecture. FEBS J 273:873–894

Le Novère N (2006) Model storage, exchange and integration. BMC Neurosci 7:S1–S11

Palsson BØ (2006) Systems biology: properties of reconstructed networks. Cambridge University Press, Cambridge

Spasić I, Simeonidis E, Messiha HL, Paton NW, Kell DB (2009) KiPar, a tool for systematic information retrieval regarding parameters for kinetic modelling of yeast metabolic pathways. Bioinformatics 25(11):1404–1411

Kinetics

- ▶ Life Span, Turnover, Residence Time
- ▶ Lymphocyte Population Kinetics

Kinetochores

Rosella Visintin
IEO, European Institute of Oncology,
Milan, Italy

Definition

Kinetochores are large proteinaceous structures that assemble on centromeres. They mediate chromosomes binding to the spindle microtubules and orchestrate sister chromatid segregation.

Cross-References

- ▶ Mitosis

k-Means

- ▶ Clustering, k-Means

Knowledge

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Definition

Knowledge is a hard term to define. Epistemology is the area of philosophy that deals with knowledge, but it has yet to yield a definition that all philosophers can agree on. Intuitively, though, most of us have a good understanding of what knowledge means. For systems biology, a working definition could be that knowledge relates to the understanding of a subject or domain.

When interpreted more computationally, knowledge can be used to refer to concepts and the relationships between concepts, or data that have been structured according to rich semantics.

In computer science, knowledge is related to information and data, and the meanings attributed to these concepts may overlap or be used interchangeably. Generally speaking, we can say that data is "raw," in the sense that it refers to the signals or information (i.e., the bits, or 0's and 1's) that derive from scientific instruments, and that are collected and stored by computers. Knowledge is more meaningful than raw data. Knowledge helps us to make informed decisions and to act in a skilful manner. An important difference between knowledge and data is that knowledge is connected to actions: knowledge is purposeful, it has meaning in the sense that if we receive some data or information and do not have any knowledge, then we do not react to the data. Whereas, if we have knowledge, we know that the information is important because we can relate it to its broader context and implications. Knowledge enables scientific discoveries to be derived from data, and it is sometimes thought of as data about data. A property of knowledge, as opposed to data, is that it can be used to generate more in the form of new insights or scientific discoveries.

Cross-References

▶ Knowledge-Based System

Knowledge Acquisition

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Definition

Knowledge acquisition covers a range of techniques used to obtain and represent knowledge about a specific area of human expertise so that it may be used in a ▶ knowledge-based system. Human experts are important sources of knowledge, and eliciting knowledge from them is a common knowledge acquisition task. Typically, the expert is interviewed by one or more knowledge engineers and asked to solve a number of case studies while attempting to outline their reasoning process. Problems arise because the expert may have developed, over many years, an instinctive or unconscious approach to problem solving, which can be very difficult to make explicit and formulize in a computational system. Nonetheless, with the help of the expert, it is usually possible for the knowledge engineers to generate a number of heuristics, or rules of thumb, which embody the ▶ knowledge of the expert and which can readily be encoded or represented in a format useful for automated inference.

Cross-References

▶ Knowledge
▶ Knowledge-based System

Knowledge Base

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Definition

A knowledge base is a collection of knowledge represented using a ▶ knowledge representation language. In systems biology, a knowledge base is used to refer to a cyber-infrastructure representing a dynamic body of scientific knowledge. The sources of knowledge contained in a knowledge base for systems biology may include:

- Important data repositories, including results from high-throughput experiments and repositories of biological models
- Software and workflow repositories that contain tools useful for data processing and analysis
- Frameworks for modeling, simulation, and making scientific predictions
- Heuristic capabilities to improve the value and sophistication of experimental design and for further scientific inquiry

In systems biology, knowledge bases are used to couple modeling and simulation to experimental design in order to facilitate further knowledge discovery. An important challenge is to combine complex and disparate sources of knowledge into a coordinated whole in order to provide scientists with a more integrated view of the various components of biological systems.

Cross-References

- ▶ Biomarker Discovery, Knowledge Base
- ▶ Knowledge
- ▶ Knowledge-based System
- ▶ Knowledge Representation

Knowledge Engineering

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Definition

Knowledge engineering is used to build information systems that utilize knowledge, concepts, and reasoning in a structured way. It involves constructing models or powerful abstractions of human ▶ knowledge about biological systems (or other areas of human endeavor). Knowledge engineering is also related to the development of computing systems for knowledge management. This involves leveraging knowledge as a key resource in scientific communities and organizations using advanced information and knowledge systems.

Cross-References

- ▶ Knowledge
- ▶ Knowledge-based System

Knowledge Inference

Luis Tari
Pharma Early Development Informatics, Hoffmann-La Roche Inc., Nutley, NJ, USA

Definition

Knowledge inference refers to acquiring new knowledge from existing facts based on certain rules and constraints. One way of representing these rules and constraints is through the use of logic rules, formally known as *knowledge representation*. The mechanism behind inferring new knowledge based on the existing facts and logic rules is typically known as *reasoning*. By encoding biological properties in the form of logic rules, the facts inferred by automated reasoning can be more biologically meaningful.

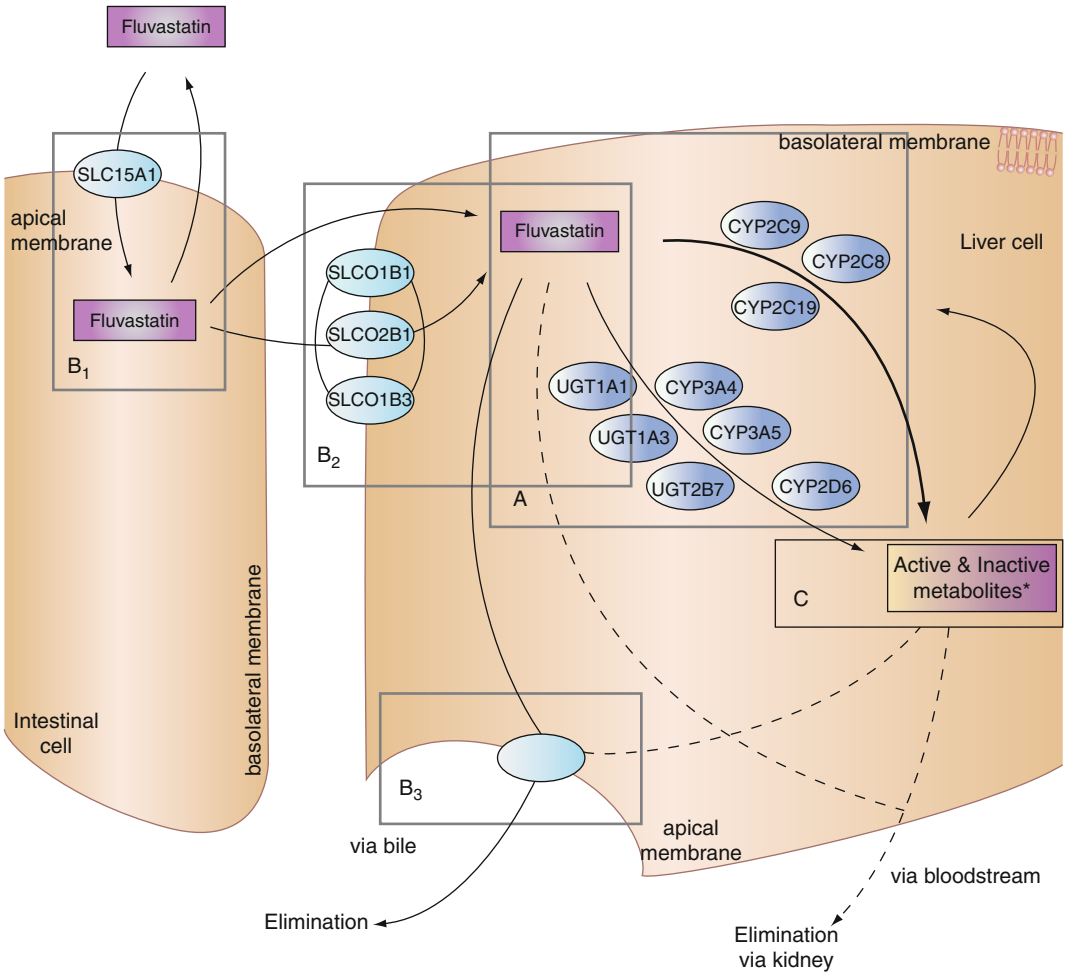
Characteristics

By acquiring biological facts from various publications, new knowledge can be inferred when relevant biological knowledge is applied. Such kind of knowledge inference enables the processing of complex tasks such as the curation of biological pathways. Knowledge inferencing can be described in two phases: (1) natural language extraction phase; (2) reasoning and inference phase. The natural language extraction phase is responsible for the acquisition of the facts from free text such as Medline abstracts.

By representing the domain knowledge in the form of logic rules, the reasoning and inference phase utilizes the extracted facts to infer knowledge. In the rest of the entry, the knowledge-inferencing mechanism is illustrated with two examples: the synthesis of pharmacokinetic pathways (Tari et al. 2010b) and the identification of drug–drug interactions (Tari et al. 2010a).

Preliminaries: Pharmacokinetics

Pharmacokinetics is concerned with the relationships between various processes during the course of the drug consumption in the body. When a drug is taken orally, the drug is absorbed in the intestine, and the



Knowledge Inference, Fig. 1 Pharmacokinetic pathway of fluvastatin. Region A: metabolism of the drug by the enzymes; Region B: drug transporters distribute the drug for absorption in intestine in B1, for metabolism in B2, and for elimination in B3; Region C: the drug is metabolized to metabolites by the enzymes (Diagram source: PharmGKB)

corresponding drug transporters move the drug into the cells via the membrane. The drug is then distributed to various organs including the liver through the bloodstream, and the relevant drug transporters in the liver cells distribute the drug for metabolism by the enzymes. Drugs that are taken intravenously would bypass the drug absorption phase. The pharmacokinetics of a drug includes several processes such as the *absorption* of a drug, the *distribution* of a drug to different tissues, the *metabolism* of a drug that leads to the conversion of the drug into metabolites, and the *elimination* of a drug (Sharif 2003). The typical processes involved in pharmacokinetic pathways are shown in Fig. 1. Drug transporters are responsible for moving the drug in (as in Regions B1 and B2 of Fig. 1) and out of the cell (as in Regions B3). Metabolism takes place when the drug is in the cell (Region A) and the drug is moved out of the cell for excretion after metabolism (Region B3). This mechanism can take place in many tissues such as intestine and liver. Once the target drug is in the cell, the enzymes play the role of metabolizing the drug (as in Region A), which take place mainly in the liver. Metabolites are produced as a result of the metabolism of the drug, shown in Region C.

Synthesis of Pharmacokinetic Pathways

Input: drug of interest

Output: pharmacokinetic pathway of the given drug

The extraction phase is performed by the PTQL (parse tree query language) extraction framework (Tari et al. 2010c), which provides the flexibility to perform diverse relationship extraction. The central piece of our extraction framework is the *parse tree database*, which is composed of the syntactic structures for each of the sentences in the entire text collection. Extraction of relationships becomes a matter of writing queries to the parse tree database.

The system gathers the necessary facts from various publications in order to synthesize pharmacokinetic pathways. Here is an overview of the extracted facts:

- *Drug–transporter distribution relations*. Before a drug can be metabolized, it is necessary for the drug to be distributed to the liver for the metabolism process. The system identifies which transporters are responsible for drug distribution. For example, `distributes(SCLO1B1, pravastatin)` corresponds to the fact that SCLO1B1 is a drug transporter for the distribution of pravastatin.
- *Drug–enzyme metabolic relations*. Enzymes play an important role in drug metabolism. The system identifies which enzymes are responsible for the metabolism of the target drug. For instance, the system represents the fact that CYP3A4 is responsible for the metabolism of pravastatin as `metabolizes(CYP3A4, pravastatin).`
- *Protein expression in liver and intestinal cells*. In the synthesis of pharmacokinetic pathways, it is essential to find out whether a protein is expressed in the liver or intestinal cells. For instance, `is_expressed(ABCC2, liver)` and `is_expressed(SLC15A1, intestine)` corresponds to the fact that ABCC2 and SLC15A1 are expressed in the liver and intestine, respectively.
- *Proteins responsible for drug elimination*. Among the interactions between the target drug and its drug transporters, it is necessary to find out the roles of each of the drug transporters, as drug transporters are known to be involved in various roles such as drug distribution, absorption, and elimination. For instance, `eliminates (ABCB1, pravastatin)` represents the fact that ABCB1 is responsible for the elimination of pravastatin.
- *Drug–metabolites relations*. The system identifies which particular metabolites are produced as a result of the metabolism of the target drug. For instance, the fact that SN-38 is a metabolite of pravastatin is represented as `metabolite_drug(SN38, pravastatin).`

The role of the reasoning phase is to represent the fundamental behavior and properties of the domain so that the extracted facts can be utilized to infer new knowledge. Implementation of the reasoning component requires a language that is ideal in specifying what kind of reasoning to be performed rather than how the reasoning is performed. AnsProlog (Gelfond and Lifschitz 1988, 1991) is a declarative language that is useful for reasoning, as well as capable for reasoning with incomplete information. The non-monotonic feature of AnsProlog allows the handling of defaults and exception.

With AnsProlog, biological properties of the relevant domain knowledge are represented in the form of logic rules. As an example of a biological property in pharmacokinetics, drug metabolites are produced as a result of drug metabolism. In other words, a metabolized drug is a precondition for the production of drug metabolites, and the effect of the interaction is the production of drug metabolites. This is represented as follows:

```
o(converts(Dr,M), Loc, T) :-
  h(metabolized(Dr, Loc),T),
metabolite_drug(M,Dr),
  metabolism_organ(Dr, Loc), not h
(converted(Dr, Loc),T).
```

The above logic rule states the preconditions for the action of converting drug Dr into metabolite M (i.e., `converts(Dr, M)`) at timepoint *T* in tissue *Loc* (which can be either the liver or intestinal cell). *Timepoints* are used to define the logical ordering of the actions involved in pharmacokinetic pathways. The following are the preconditions, which are specified to the right of the "if" symbol `:-` in the rule, for the action `converts(Dr, M)`:

- The drug D has been metabolized in tissue Loc at timepoint T, denoted as `h(metabolized(Dr,Loc),T)`
- Metabolite M is known to be a metabolite of Dr, denoted as `metabolite_drug(Dr, M)`

- Metabolism of D is known to take place in Loc, denoted as
  `metabolism_organ(Dr, Loc)`
- It is not known that D has been converted into metabolites in tissue Loc in the previous timepoints, denoted as
  `not h(converted(Dr, Loc),T)`

By encoding the logic representation in the form of pre- and post-conditions of the actions involved in the consumption of drugs, reasoning can be applied to find an explanation of how the drug is consumed based on the extracted facts.

Suppose we are interested in the pharmacokinetic pathway of fluvastatin, we provide the following logic facts as input:

```
drug(fluvastatin)
h(is present(fluvastatin,
  intestine),0)
```

The above logic fact indicates that fluvastatin is consumed and present in the intestine.

The reasoning component takes the extracted logic facts from the extraction phase and the logic rules representing properties of pharmacokinetics to infer the following sequence of actions that lead to the consumption of fluvastatin:

```
o(distributes(slco1b1,fluvastatin),
  liver,1)
o(metabolizes(cyp1a1,fluvastatin),
  liver,2)
o(metabolizes(cyp2c8,fluvastatin),
  liver,2)
o(metabolizes(cyp2c9,fluvastatin),
  liver,2)
o(metabolizes(cyp3a4,fluvastatin),
  liver,2)
o(metabolizes(cyp2c19,fluvastatin),
  liver,2)
o(metabolizes(cyp2d6,fluvastatin),
  liver,2)
o(converts(fluvastatin,x6 hydroxy),
  liver,3)
o(converts(fluvastatin,m2),liver,3)
o(converts(fluvastatin,m3),liver,3)
o(converts(fluvastatin,m4),liver,3)
o(converts(fluvastatin,m7),liver,3)
o(eliminates(abcb1,fluvastatin),
  liver,4)
```

Timepoints are used to define the logical ordering of the actions involved in pharmacokinetic pathways. Action A_1 occurs before interaction A_2 if A_1 is assigned with a timepoint that is smaller than the timepoint for A_2. For instance, the timepoints indicate that the action *metabolizes* occurs ahead of the action *converts*.

Preliminaries: Drug–Drug Interactions

The issue of drug–drug interactions has received great amount of attention as it may cause adverse drug reactions. Drug–drug interactions are concerned with how the consumption of a drug is influenced by another drug. Inhibition of enzymes is a common form of drug–drug interactions (Boobis et al. 2009). This kind of *direct inhibition* happens when drug *A* inhibits enzyme *E*, which is responsible for the metabolism of drug *B*. The inhibition by *A* leads to the decrease of the activity level of *E*, and this in turn may delay the disposition of drug *B*. Such unexpected delay can potentially lead to side effects for patients who are administered with drugs *A* and *B*. An example of such direct inhibition is the interaction between quinidine and CYP2D6 substrates such as codeine. Quinidine is responsible for the inhibition of the CYP2D6 enzyme, while codeine is metabolized by CYP2D6. The inhibition of CYP2D6 by quinidine can increase the effect of codeine. Such increase can potentially lead to adverse side effects of the affected drug.

Another form of drug interactions is through the induction of enzymes (Boobis et al. 2009). One form of induction is known as *direct induction* when drug *A* induces enzyme *E*, which is responsible for metabolism of drug *B*. An example of such direct induction is between warfarin and phenobarbital. Such drug interaction occurs due to the fact that warfarin is metabolized by the CYP2C9 enzyme, while CYP2C9 is subject to induction by phenobarbital. This leads to the increase of enzyme activity of CYP2C9, which increases the rate of metabolism of warfarin by CYP2C9. Such increase of metabolism decreases the life span of warfarin. While direct induction is possible, it is not the most common form of drug interactions due to induction. A more common form is through transcription factors that regulate the drug-metabolizing enzymes. An alternative form is *indirect induction* through transcription factors. Such interaction occurs when drug *A* activates transcription factor *TF*, which regulates and induces enzyme *E*, and enzyme *E* is responsible for the metabolism of drug *B*.

Such transcription factors are referred as regulators for xenobiotic-metabolizing enzymes. There are other kinds of mechanisms that lead to drug interactions, but here we focus on these three types.

Identifying Drug–Drug Interactions

Input: drug of interest

Output: potential interaction between a drug and drug of interest

In drug design, it is important to identify potential drug interactions in the design process. Given that the effect of the drug is known as an inducer or inhibitor for an enzyme, the question is to identify drugs that can be affected by this new drug. Suppose our new drug is a CYP3A4 inhibitor, we represent such information in the form of logic facts as the input to our system.

```
drug(new_drug). enzyme(cyp3a4).
  inhibits(new_drug, cyp3a4).
```

Among the extracted facts, terfenadine is found to be one of the drugs that are metabolized extensively by CYP3A4. This is supported by the following evidence sentence:

Testosterone, terfenadine, *midazolam, and nifedipine, four commonly used substrates for human* cytochrome P-450 3A4 (CYP3A4) (PMID: 10681383)

This leads to the logic fact `metabolized (terfernadine, cyp3a4)`. With the input, the extracted facts, and the logic rules, the reasoning component returns the following answer set:

```
affects(new_drug, level(cyp3a4, low))
result(new_drug,          increases,
   terfenadine)
```

The answer set indicates that the new drug may increase the effect of terfenadine, since the new drug decreases the expression activity of CYP3A4.

References

Boobis A, Watelet J-B, Whomsley R, Benedetti MS, Demoly P, Tipton K (2009) Drug interactions. Drug Metab Rev 41(3):486–527

Gelfond M, Lifschitz V (1988) The stable model semantics for logic programs. In: International symposium on logic programming, Cambridge, Massachusetts, pp 1070–1080

Gelfond M, Lifschitz V (1991) Classical negation in logic programs and disjunctive databases. New Generat Comput 9:365–387

Sharif ZA (2003) Pharmacokinetics, metabolism, and drug-drug interactions of atypical antipsychotics in special populations. J Clin Psychiatry 5:22–25

Tari L, Anwar S, Liang S, Cai J, Baral C (2010a) Discovering drug interactions: a text-mining and reasoning approach based on properties of drug metabolism. In: Proceedings of the 9th European conference on computational biology (ECCB 2010), Sept 2010, Gent, Belgium

Tari L, Anwar S, Liang S, Hakenberg J, Baral C (2010b) Synthesis of pharmacokinetic pathways through knowledge acquisition and automated reasoning. In: Proceedings of the pacific symposium on biocomputing (PSB'10), Gent, Belgium

Tari L, Tu PH, Hakenberg J, Chen Y, Son TC, Gonzalez G, Baral C (2010c) Incremental information extraction using relational databases. To appear in IEEE transactions on knowledge & data engineering (TKDE), Gent, Belgium

Knowledge Management

Ulf Leser[1] and Wolfram Liebermeister[2]

[1]Institute for Computer Science, Humboldt-Universität zu Berlin, Berlin, Germany

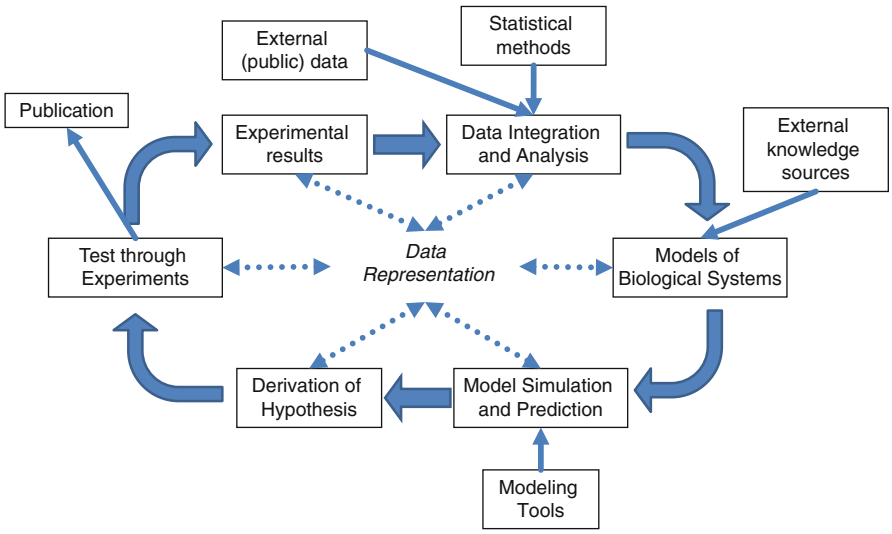
[2]Institut für Biochemie, Charité-Universitätsmedizin Berlin, Berlin, Germany

Synonyms

Data management; Information integration; Information management; Knowledge organization

Definition

Knowledge Management (KM) encompasses techniques and processes to represent, store, search, integrate, and analyze ▶ knowledge that is available in digital form. Knowledge Management in Systems Biology (KMSB) is concerned with knowledge about biological systems and the way they interact to sustain life. Specifically, KMSB covers the problems of formally representing knowledge about biological entities and processes, the robust generation of biological knowledge from raw experimental data, the integration of knowledge from multiple and heterogeneous data sources, and the provision of search and inference algorithms to access and use the knowledge being managed by a system. Since human language is especially appropriate to represent complex knowledge but inadequate for being used by computers, an important



Knowledge Management, Fig. 1 The cyclic process of experiment and analysis

subtask of KMSB is the transformation of knowledge that is represented in unstructured text into structured representations by ▶ text mining.

Characteristics

The dominant aim of Systems Biology is to model biological systems. To this end, researchers need to infer knowledge, for instance, about the structure and dynamics of ▶ biochemical pathways, from raw experimental data, like measured protein or metabolite levels. Knowledge Management for Systems Biology (KMSB) is concerned with developing algorithms and tools that enable this work in a collaborative fashion. The term does not denote a specific method, but rather encompasses a range of different techniques. KMSB transfers many results from research in information systems, ▶ data mining, collaboration, and ▶ information integration to the world of biological modeling and simulation. In addition, it addresses the specific problems of Systems Biology, which include the enormous differences in the scale of the objects being studied (from single molecules to a complete living being), the extreme heterogeneity in the nature of the raw data used, the complexity of the interplay between the entities involved, and our limited ability to measure or observe specific aspects of these systems.

In Knowledge Management, it is helpful to distinguish between data (raw, numerical, derived from experimentation), information (data in context; raw data interpreted in the context of a biological question), and knowledge (abstract and confirmed set of information answering a biological question). While KMSB is mostly concerned with the latter, it also touches upon the formers since all levels need to be considered to ensure reliable knowledge. However, research in KMSB is mainly focused on knowledge encoded in or needed for building models of biological systems.

Cycle of Experiment and Analysis

Figure 1 shows the prototypical process of model development. Systems Biologists produce and integrate various types of data, including properties of gene sequences, high-throughput ▶ transcriptome, ▶ metabolome, or ▶ proteome data sets, or molecular interaction networks. After collecting such data and processing them by statistical data analysis, researchers develop abstract ▶ mathematical models of the biological system under study. Developing such models requires, apart from the experimental data, input from other sources including textbooks, the scientific literature, existing models, and human experts. Model simulations are then used to derive new hypotheses about the modeled system, which must be validated or falsified by novel experiments, leading to new insights and refined models. Important original data and modeling results are published and possibly stored in specific databases. KMSB has to support all steps of this process.

Knowledge Management, Table 1 Standards for data exchange and representation

Name	Scope	URL
Gene ontology	Ontology for cell biological objects and functions	www.geneontology.org/
Minimum information for biological and biomedical investigations	Collection of minimal requirements (for metadata, etc.)	mibbi.org/
Minimum information required in the annotation of models	Minimum requirements for describing biochemical models	www.ebi.ac.uk/miriam/
Systems biology markup language	Computational models	www.sbml.org/
Systems biology graphical notation	Network diagrams	sbgn.org/
BioPAX	Language for exchanging biochemical pathways	www.biopax.org/
PSI MI	Molecular interactions	psidev.sourceforge.net/

Knowledge Representation

KMSB faces a variety of data, including experimental results, biochemical knowledge, and computational models. As a requirement for the exchange of data and tools acting upon this data, these need to be represented in standardized, computer-readable formats. Formats like ▶ SBML (Systems Biology Markup Language, Hucka et al. 2003) cover different application domains and are used as exchange formats by Systems Biology databases. Biochemical networks can be represented conveniently by the diagrammatic language ▶ SBGN (Systems Biology Graphical Notation), a standard for representing network graphics. Furthermore, every data set needs to be enriched with meta-information to allow its proper usage. Conventions for minimal sets of metadata for various sorts of data are coordinated by the MIBBI effort (Taylor et al. 2008). A prominent example, the MIRIAM standard for computational models (Le Novère et al. 2005), is aimed to ensure a reliable documentation of models and, ultimately, reproducibility of simulation results. A list of important standards in Systems Biology may be found in Table 1.

Data exchange also requires a standardized naming of fundamental objects (like proteins or metabolites) and basic concepts (like functions of a protein). Ontologies like the ▶ gene ontology or the ▶ systems biology ontology provide both controlled vocabularies and definitions of biochemical concepts like enzymatic mechanisms, gene functions, or the sub-cellular location of proteins. Elements of a computational model can be linked to ontology elements by so-called annotations, which greatly help to search data sets effectively and to interlink heterogeneous data from different sources. Furthermore, annotations can encode biochemical knowledge to be used for analysis, e.g., to correlate the expression and the biological function of genes.

Knowledge Extraction

Research projects rarely start on entirely new topics. Therefore, existing knowledge from textbooks, scientific publications, and biological databases play an important role in developing and improving Systems Biology models. Accessing this knowledge can be anything from trivial to nearly impossible. If knowledge is available in structured form, i.e., in databases, accessing it requires efficient search methods and an understanding of formats and names (see below). However, novel knowledge is still predominately published in articles, i.e., in natural English texts; in the same way, established knowledge is still published primarily in text books. Extracting knowledge from text can be performed manually or automatically using ▶ text mining:

- Many large Systems Biology databases (e.g., Brenda, IntAct, or Sabio-RK) employ human curators who read selected texts and transform the relevant information from the text into a computer-readable format. Although curated databases are generally considered to contain high-quality data, this view is sometimes questioned, especially in terms of completeness (Cusick et al. 2009). Therefore, most Systems Biology researchers perform their own literature searches during modeling. Such searches can be supported by automated tools specializing in biomedical information retrieval (Hoffmann et al. 2005).
- A second approach to knowledge extraction from text is to scan the texts automatically by computer programs using biomedical text mining. It involves a series of subtasks including text preprocessing,

grammatical annotation, identification of biological entities (genes, chemicals, cell lines etc.), and extraction of relationships between these entities (regulation, complex formation, metabolic reactions etc.). A number of existing systems employ such methods to reconstruct the topology of biological networks (Bauer-Mehren et al. 2009). Another line of research targets the extraction of ▶ kinetic parameters or the stoichiometry of reactions (▶ Stoichiometric Mass Balance Analysis) (Hakenberg et al. 2004).

Computational biological models are a particularly complex form of knowledge and creating them requires extensive efforts in knowledge collection. Many large-scale efforts to reconstruct networks (e.g., by Reactome, KEGG, BioCyc, Yeast consensus model) mainly target the topology of networks by listing and annotation of relevant compounds and reactions. Computational procedures such as ▶ flux analysis can be used to validate that the reconstructed networks are biochemically plausible. An alternative approach to collecting and distributing knowledge is provided by wikis (e.g., Lammers et al. 2010). In contrast to traditional databases, wikis do not impose a fixed data structure, which makes it difficult to check the accumulated knowledge for consistency. However, when various pieces of information need to be collected from many domain experts, the easy access provided by wikis turns out to be a strong advantage.

Knowledge Integration

Knowledge in Systems Biology is scattered over dozens of different data sources, including protein databases like UniProt, ▶ model databases such as BioModels or JWS online, databases on enzyme biochemistry such as Brenda or Sabio-RK, or repositories of experimental results such as GEO or ArrayExpress. Acquiring a comprehensive view on these diverse databases or performing a joint analysis of different data sets requires their integration. This task is typically divided in two subtasks: Information Integration is concerned with heterogeneity in metadata, i.e., formats or database schemas (Stein 2008). Solutions to these problems are mostly built upon exchange standards as described above. Data integration, in contrast, addresses the biological data and is concerned with statistical data normalization and experimental biases (Searls 2005). By integrating data within quantitative models, new knowledge may be inferred. For instance, large ▶ metabolic network models can help to validate and complete ▶ thermodynamic data sets. A pertinent problem in data integration is the usage of different identifiers. To interrelate different data sets or to map experimental data onto models, data elements need to be aligned to each other. This is usually achieved by matching their annotations, which point to public databases and ontologies like KEGG, ChEBI, Uniprot, or the Gene Ontology. Comparing the annotations is also necessary to align computational models and can help to spot inconsistencies between them (Krause et al. 2009). Jamborees, targeted meetings of domain experts, have turned out to be an efficient way to obtain network integration in a community effort (Herrgard et al. 2008).

A number of databases have taken over the task of providing integrated data sets (e.g., Chowbina et al. 2009). These integrated databases relieve a user from performing integration herself, but usually carry the danger of being outdated or of missing some specific information. Recently, scientific workflow engines have been proposed as the appropriate mean to implement custom-data integration algorithms in a clean, reusable, and extensible manner (Li et al. 2010).

Cross-References

- ▶ Bio-Ontologies
- ▶ Data Mining
- ▶ Gene Ontology
- ▶ Knowledge
- ▶ Text Mining

References

Bauer-Mehren A, Furlong LI, Sanz F (2009) Pathway databases and tools for their exploitation: benefits, current limitations and challenges. Mol Syst Biol 5:290

Chowbina SR, Wu X, Zhang F, Li PM, Pandey R, Kasamsetty HN, Chen JY (2009) HPD: an online integrated human pathway database enabling systems biology studies. BMC Bioinformatics 10(Suppl 11):S5

Cusick ME, Yu H, Smolyar A, Venkatesan K, Carvunis AR, Simonis N, Rual JF, Borick H, Braun P, Dreze M et al (2009) Literature-curated protein interaction datasets. Nat Methods 6(1):39–46

Hakenberg J, Schmeier S, Kowald A, Klipp E, Leser U (2004) Finding kinetic parameters using text mining. OMICS - J Integr Biol 8(2):131–152

Herrgard MJ, Swainston N, Dobson P, Dunn WB, Arga KY, Arvas M, Blüthgen N, Borger S, Costenoble R, Heinemann M et al (2008) A consensus yeast metabol-ic network

reconstruction obtained from a community approach to systems biology. Nat Biotechnol 26(10):1155–1160
Hoffmann R, Krallinger M, Andres E, Tamames J, Blaschke C, Valencia A (2005) Text mining for metabolic pathways, signaling cascades, and protein networks. Sci STKE 283:pe21
Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A et al (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics 19(4):524–531
Krause F, Uhlendorf J, Lubitz T, Schulz M, Klipp E, Liebermeister W (2009) Annotation and merging of SBML models with semanticSBML. Bioinformatics 26(3):421
Lammers CR, Florez LA, Schmeisky AG, Roppel SF, Mäder U, Hamoen L, Stülke J (2010) Connecting parts with processes: SubtiWiki and SubtiPathways integrate gene and pathway annotation for Bacillus subtilis. Microbiology 156(Pt 3):849–859
Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, Collado-Vides J, Crampin EJ, Halstead M, Klipp E, Mendes P et al (2005) Minimum information requested in the annotation of biochemical models (MIRIAM). Nat Biotechnol 23(12):1509–1515
Li P, Dada JO, Jameson D, Spasic I, Swainston N, Carroll K, Dunn W, Khan F, Malys N, Messiha HL et al (2010) Systematic integration of experimental data and models in systems biology. BMC Bioinformatics 11(1):582
Searls DB (2005) Data integration: challenges for drug discovery. Nat Rev Genet 4:45–58
Stein LD (2008) Towards a cyberinfrastructure for the biological sciences: progress, visions and challenges. Nat Rev Genet 9(9):678–688
Taylor CF, Field D, Sansone SA, Aerts J, Apweiler R, Ashburner M, Ball CA, Binz PA, Bogue M, Booth T et al (2008) Promoting coherent minimum reporting guidelines for biological and biomedical investigations: the MIBBI project. Nat Biotechnol 26(8):889–896

Knowledge Organization

▶ Knowledge Management

Knowledge Representation

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Definition

Knowledge representation refers to the technical problem of encoding human ▶ knowledge and reasoning (▶ Automated Reasoning) into a symbolic language that enables it to be processed by information systems. In systems biology, knowledge representation is used to infuse data with scientific concepts and understanding in order to maximize its utility for furthering scientific insight.

Characteristics

Knowledge representation is an active area of research in artificial intelligence (Brachman and Bector 2004). It often refers to the complex and time-consuming technical process performed by knowledge engineers (▶ Knowledge Engineering) when acquiring domain knowledge for use in ▶ knowledge-based systems.

The question of how to represent human ▶ knowledge is an old problem, and knowledge representations are not limited to the rule-based approaches (▶ Rule-based Methods; ▶ Rule Discovery) typically associated with ▶ knowledge-based systems. Among the first knowledge representations are prehistoric bones, carved with notches that enumerate the lunar phases. Other knowledge representations include hieroglyphics, writing, arithmetic, mathematical tables, and algebra, as well as various methods and languages developed by artificial intelligence researchers to reproduce human cognitive processes in computers. Knowledge representations are closely related to methods of reasoning: the purpose of a knowledge representation, especially in the context of systems biology, derives from its practical value in allowing us to make predictions about the world.

Davis et al. (1993) argue that the fundamental task of knowledge representation is to describe the natural world. They elucidate five important roles for a knowledge representation:

1. Knowledge representations are a substitute or surrogate of the world. All representations or models are inherently inaccurate because they are simplifications of the real thing and may contain artifacts.
2. Knowledge representations are a set of ontological commitments. Knowledge representations approximate reality: different representations describe reality from different points of view. The decisions made about what is important for the representation and therefore should be included, and what can be left out, are ontological.

3. Knowledge representations are fragmentary theories of intelligent reasoning. Sets of rules and sets of mathematical equations are two different knowledge representations that can be used to make predictions but which must be used with different systems of reasoning. A knowledge representation encourages us to reason about the world in a certain way, and it restricts the types of conclusion that we can draw.
4. Knowledge representations enable helpful or efficient ways of computing and making predictions about the world. If they did not, then there would be no point in using them.
5. A knowledge representation is a human language. It enables us to express things about the world, and it acts a medium of communication.

Examples of knowledge representations that are commonly used with ▶ knowledge-based systems include:

- Symbolic logic: Also known as predicate logic or first-order logic. Here, the term *predicate* is used to refer to the *relationships* between objects, where an object could be a physical object like a biological organism, or a concept like a measurement of some property of an organism. The objects are then called the *arguments* or *terms* of the predicate, and the values of the arguments are allowed to vary so that a predicate may be said to be true or false depending on the values of the arguments. In a ▶ knowledge base, predicates that are logically true are used to represent a collection of facts. Computer languages like ▶ PROLOG, which are based on predicate logic, are able to automate logical processes like ▶ deduction and ▶ induction and are able to discover new ▶ knowledge.
- Heuristics: Rules that guide a search in a particular problem space. They may be represented in an IF-THEN format, meaning that if conditions are satisfied then a certain action may be performed or a conclusion may be drawn.
- Frames: Are similar to objects in object-orientated programming. Frames have slots instead of attributes and procedural attachments instead of methods. They enable ▶ knowledge to be structured in a hierarchical way and can be used to describe various biological concepts such as reactions, pathways, and physical entities such as molecules. For example, in Vastrik et al. (2007), a particular reaction may be represented as an instance of the frame "reaction," and the frame may have slots for inputs and outputs, which respectively represent the reactants and products of the given reaction.
- Semantic networks: Graphs that use nodes to store facts and links to represent relations between facts. They were originally designed to model human memory, with concepts stored at the nodes of the graphs, and links that related nodes to each other.

What is important for a valid knowledge representation is that its formal structure is representational and carries meaning: It is not simply a data structure. A knowledge representation creates a correspondence between the constructs of the representation and the real world, a data structure does need such a correspondence. In this sense, data is just data; it carries no extra meaning and does not need to be related to its context in a wider world.

Data structures, however, are used to implement knowledge representations. For example, a knowledge representation such as a semantic network may be stored using a graphical data structure. The graph itself has no correspondence with the domain of knowledge being represented: It is the semantics or meaning attached to the graph that defines this correspondence and that dictates the graph's topology.

Knowledge representations in systems biology are often used to represent both qualitative and quantitative scientific ▶ knowledge. Such knowledge representations are based on community-agreed standardized tools and formats (Brazma et al. 2006; Wang et al. 2005). Some examples of important standardized formats for knowledge representation in systems biology include:

- XML-based languages such as SBML: The Systems Biology Markup Language for representing stoichiometric and regulatory models; BioPAX: Biological pathway exchange language to enable the integration, exchange, visualization, and analysis of biological pathway data; and CellML: to store, share, and exchange mathematically based models of cells.
- Collections of computational models such as the BioModels Database and the related MIRIAM standard. MIRIAM is an effort to help the systems biology community to collaborate when annotating quantitative models of biological systems with scientific knowledge.

- ▶ Ontologies from the Open Biological and Biomedical Ontologies (OBO) initiative. These include the Gene Ontology (GO) and the Protein Ontology (PRO). Ontologies are data models concerned with conceptualization and embody a community's knowledge of a domain (Bard and Rhee 2004). They define the basic terms and relations of a domain of interest, as well as the rules for combining these terms and relations so that they may be used for reasoning. Standardized ontologies are important for curating and annotating key scientific data sets (e.g., reference genome sequences).

Knowledge representation in artificial intelligence has typically focused on narrow domains of human expertise, and this has been a crucial factor for developing successful knowledge-based applications. In contrast, systems biology attempts to look at the bigger picture. It adopts a holistic approach to science, as opposed to a reductionist approach, and it therefore inherently attempts to incorporate ▶ knowledge from a wide variety of domains and scientific disciplines. The application of knowledge representation and knowledge-based technologies to systems biology is therefore challenging. In addition, with the rapid growth in biological data repositories and the rapid development of new technologies for measuring biological systems, the need for standardized systems of knowledge representation continues to grow. The development and widespread adoption of such standards by the community is of great importance: They are crucial for integrating diverse systems of scientific ▶ knowledge into a coherent whole.

Cross-References

▶ Automated Reasoning
▶ Deduction
▶ Induction
▶ Knowledge
▶ Knowledge Base
▶ Knowledge Engineering
▶ Knowledge-based System
▶ PROLOG

References

Bard JBL, Rhee SY (2004) Ontologies in biology: design, applications and future challenges. Nat Rev Genet 5:213–222
Brachman RJ, Levesque BJ (2004) Knowledge representation and reasoning. Morgan Kaufmann, Amsterdam
Brazma A, Krestyaninova M, Sarkans U (2006) Standards for systems biology. Nat Rev Genet 7:593–605
Davis R, Shrobe H, Szolovits P (1993) What is a knowledge representation? AI Mag 14(1):17–33
Vastrik I, D'Eustachio P, Schmidt E, Joshi-Tope G, Gopinath G, Croft D, de Bono B, Gillespie M, Jassal B, Lewis S, Matthews L, Wu G, Birney E, Stein L (2007) Reactome: a knowledge base of biologic pathways and processes. Genome Biol 8:R39
Wang X, Gorlitsky R, Almeida JS (2005) From XML to RDF: how semantic web technologies will change the design of 'omic' standards. Nat Biotechnol 23:1099–1103

Knowledge-based System

Martin Swain
Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University, Aberystwyth, Ceredigion, UK

Synonyms

Expert system

Definition

A knowledge-based system is a computer program that uses a ▶ knowledge base with an inference engine in order to solve problems that usually require significant specialized human expertise. It embodies the problem-solving ▶ knowledge of a human expert in a narrowly defined domain and it is able to extend that body of ▶ knowledge through its inference engine or query system.

Characteristics

People who possess specialized ▶ knowledge of a particular domain are called experts. This knowledge may take the form of a set of rules or heuristics about how to deal with particular circumstances and problems. Expert knowledge can be embodied in a computer through techniques of ▶ knowledge acquisition, ▶ knowledge representation, and ▶ knowledge engineering: a ▶ knowledge base is a collection of such knowledge. Knowledge-based systems combine

a knowledge base with automated inference mechanisms: they replicate the cognitive processes that are employed by experts when solving problems in a specific area of human endeavor. An important aspect of knowledge-based systems is that they are able to outline the reasoning process by which they reached their conclusions. They are one of the first successful applications of Artificial Intelligence (Giarratano and Riley 1998; Negnevitsky 2002; Liao 2004).

The key components of a knowledge-based system are a ▶ knowledge base and an automated inference mechanism. The inference or reasoning system uses human ▶ knowledge that has been encoded in the knowledge base to infer new beliefs and new knowledge. One of the earliest examples of knowledge-based systems is the program Dendral, which used heuristics derived from the reasoning process of organic chemists to identify organic molecules from mass spectra and chemical knowledge. A more modern example is Adam (King et al. 2009), a laboratory robot that is able to independently perform experiments in yeast-based functional genomics. It is able to test hypotheses and interpret findings without human guidance. Adam is claimed to be the first machine in history to have discovered new scientific knowledge independently of its human creators.

An important characteristic of knowledge-based systems is that there is a clean separation between the ▶ knowledge base and the inference mechanism. Knowledge-based systems adopt the paradigm of declarative programming, in which the goal of a program is separated from the methods used to achieve it. In declarative programming, the idea is that the user specifies the goal (i.e., a hypothesis to be tested), and the underlying inference mechanisms of the knowledge-based system then try to achieve that goal. Various expert system shells are available to help with the development of knowledge-based systems. Expert system shells come with inbuilt reasoning mechanisms, but they do not contain any ▶ knowledge: it is up to the knowledge engineers to enter knowledge into the shell using a suitable representation [▶ knowledge representation].

A ▶ knowledge base may represent domain ▶ knowledge as a set of rules in the IF (condition, or pattern to match) THEN (consequent, or action) structure (▶ Rule). When the condition part of a rule is satisfied, then the rule is said to "fire," and the action part is executed. The inference engine carries out the process of automated reasoning that allows the knowledge-based system to reach a solution. The inference engine instantiates rules with facts. Usually, the inference engine has a prioritized list of rules to satisfy first.

An example of a rule from the MYCIN system for the diagnosis of meningitis and bacteremia (bacterial infections) is as follows:

IF the site of the culture is blood, and the identity of the organism is not known with certainty, and the stain of the organism is gram-negative, and the morphology of the organism is rod, and the patient has been seriously burned

THEN there is weakly suggestive evidence (0.4) that the identity of the organism is pseudomonas

Methods of inference used by knowledge-based systems in which knowledge is represented as rules (▶ Knowledge Representation) include forward chaining and backward chaining. Chains of inference are created as a consequence of a rule firing. When a rule fires a new fact may be derived, which is then added to the fact database (the set of facts currently in working memory). This new fact could then cause additional rules to fire. In this way, chains of interference are created. An important way to control the firing of rules is to prioritize certain rules, so that if many rules are able to fire at the same time, then the prioritized rules will fire first. Each rule may only fire once in order to avoid problems associated with endless loops of circular reasoning.

Forward chaining is data-driven reasoning. It begins with known facts or data that cause rules to fire. Firing rules may add more data to the set of facts in working memory, and the process of forward chaining continues until no more rules may be fired. Forward chaining is concerned with drawing conclusions or interpreting something of interest from a given set of facts. It is useful for prognosis, monitoring, and control.

Backward chaining is goal-driven reasoning. It is concerned with testing hypotheses or achieving goals. It starts with a hypothesis, assumes the hypothesis is true, and then traces backward through the stack of rules to find a set of facts that support the hypothesis. Whereas in forward chaining the inference engine begins with facts and the associated IF parts of rules, in backward chaining the inference engine begins with the actions or THEN parts of rules. To fire a rule, the inference engine must first meet the condition part (IF part) of the rule. It may only be able to do this by, for example, adding a new fact to the fact database – which might be the goal or action part of another rule. By first looking at the THEN parts of the rules in order to identify rules of interest, and only

then considering how to satisfy the condition or IF part of these rules, inference chains growing backward through the set of rules are created. Backward chaining is typically used for diagnostic problems.

The types of reasoning and inference performed by the knowledge-based system depend on the manner in which knowledge is represented (▶ Knowledge Representation) in the ▶ knowledge base. Knowledge-based systems for systems biology may consist of a simulation environment that can import and execute mathematical models described using SBML, along with a rule-based decision support system that uses ontologies to gain knowledge of different physical entities. An example of such a system could be to understand the medical problems associated with tissues or organs in the human body.

Symbolic computing is a common methodology used with knowledge-based systems. Prolog (▶ PROLOG) is an example of a symbolic computing language based on predicate logic. Symbolic computing encodes qualitative data, such as words, phrases, and sentences, into symbols. For example, a ▶ knowledge base might consist of a collection of propositions represented as symbols. Through the mechanisms of logical reasoning and the formal manipulation of symbols, the knowledge-based system is able to derive symbolic representations of new propositions. The new propositions are assumed to be true and can be used for further reasoning or decoded into words and phrases that may be presented to a user as a recommended course of action.

Knowledge-based systems may be demanding to develop. They must be created individually for different application domains. There is no general technique for verifying the completeness and consistency of knowledge-based systems, which makes it difficult to identify incomplete or inconsistent knowledge. Also, because they are restricted to very narrow areas of human expertise, they may be inflexible. They cannot be applied to general problems.

Cross-References

- ▶ Biomarker Discovery, Knowledge Base
- ▶ Knowledge
- ▶ Knowledge Acquisition
- ▶ Knowledge Base
- ▶ Knowledge Engineering
- ▶ Knowledge Representation
- ▶ PROLOG
- ▶ Rule

References

Giarratano J, Riley G (1998) Expert systems: principles and programming, 3rd edn. PWS Publishing, Boston. ISBN 0-534-95053-1

King RD, Rowland J, Oliver SG, Young M, Aubrey W, Byrne E, Liakata M, Markham M, Pir P, Soldatova LN, Sparkes A, Whelan KE, Clare A (2009) The automation of science. Science 324(5923):85–89

Liao SH (2004) Expert system methodologies and applications – a decade review from 1995 to 2004. Expert Syst Appl 28(1):93–103

Negnevitsky M (2002) Artificial intelligence: a guide to intelligent systems, 1st edn. Addison-Wesley, New York, An imprint of Pearson Education. ISBN 0-201-71159-1

Koch's Molecular Postulates

▶ Koch's Postulates

Koch's Postulates

Christian Schönbach
Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, Iizuka, Fukuoka, Japan

Synonyms

Koch's molecular postulates; Koch's postulates

Definition

Robert Koch presented at the Tenth International Congress of Medicine in Berlin in 1890 three conditions that must be fulfilled to proof the causal relationship between a microbe and a disease:

1. The parasite occurs in every case of the disease in question, and under circumstances which can account for the pathological changes and clinical course of the disease.
2. The parasite occurs in no other disease as a fortuitous and nonpathogenic parasite.

3. After being fully isolated from the body and repeatedly grown in pure culture, it can induce the disease anew. (Modified after Thomas Rivers' English translation of Koch's postulates [Rivers 1937])

The third postulate poses a challenge for PCR and sequence-based identification of new viruses and bacteria. In 1988 Falkow proposed modifications, called molecular Koch's postulates that consider the association of a gene and pathogenicity as necessary, rather than sufficient (Falkow 1988; Fredericks and Relman 1996).

Cross-References

▶ Koch's Postulates
▶ Viral Respiratory Tract Infections

References

Falkow S (1988) Molecular Koch's postulates applied to microbial pathogenicity. Rev Infect Dis 10(Suppl 2):S274–S276
Fredericks DN, Relman DA (1996) Sequence-based identification of microbial pathogens: a reconsideration of Koch's postulates. Clin Microbiol Rev 9(1):18–33
Rivers TM (1937) Viruses and Koch's postulates. J Bacteriol 33(1):1–12

Kolmogorov Forward Equation

▶ Stochastic Processes, Fokker-Planck Equation

Kozak Consensus

▶ Kozak Consensus Sequence

Kozak Consensus Sequence

Leoš Shivaya Valášek
Laboratory of Regulation of Gene Expression, Institute of Microbiology AVCR, Prague, Czech Republic

Synonyms

Kozak consensus; Kozak sequence

Definition

The Kozak consensus sequence (gcc)gccRccAUGG, where R is a purine (adenine or guanine) three bases upstream of the start codon (AUG), which is followed by another "G" (Kozak 1986). This sequence on an mRNA molecule is recognized by the ribosome as the translational start site, from which a protein is produced according to the coding template of a gene carried on that mRNA molecule. In vivo, this site is often not matched exactly on different mRNAs and the amount of protein synthesized from a given mRNA is dependent on the strength of the Kozak sequence. The AUG triplet is the most important because in the vast majority of cases it is the actual initiation codon encoding a methionine amino acid at the N-terminus of each protein. The A nucleotide of the "AUG" triplet is referred to as number +1. For a "strong" consensus, the nucleotides at positions +4 (i.e., G in the consensus) and -3 (i.e., either A or G in the consensus) relative to the number 1 nucleotide must match the consensus.

References

Kozak M (1986) Point mutations define a sequence flanking the AUG initiator codon that modulates translation by eukaryotic ribosomes. Cell 44:283–292

Kozak Sequence

▶ Kozak Consensus Sequence

Kullback-Leibler Distance

▶ Kullback-Leibler Divergence

Kullback-Leibler Divergence

Daniel Polani
Adaptive Systems Research Group, School of Computer Science, University of Hertfordshire, Hatfield, UK

Synonyms

Kullback-Leibler distance

Definition

Measure of dissimilarity between two ▶ probabilistic variables defined over the same set of outcomes.

Formally, let X and X' be two probabilistic variables with the same set of outcomes χ and with associated probabilities p and p'. Then, the Kullback-Leibler divergence $D(X'||X)$ (Cover and Thomas 2006) of the variables is defined by

$$D(X'||X) \equiv D(p'||p) \tag{1}$$

$$:= \sum_{x\in\chi} p'(x) \log \frac{p'(x)}{p(x)}. \tag{2}$$

Special Cases: When probabilities can vanish in the sum, use the convention of $0 \log 0 \equiv 0$ and $0 \log \frac{0}{0} \equiv 0$ (see also ▶ Entropy). If for one of the summands $p(x) = 0$, but $p'(x) \neq 0$, set $D(X'||X) := \infty$.

Characteristics

The Kullback-Leibler divergence provides either a nonnegative real value or infinity. It becomes 0 exactly when the probability measures p and p' for X and X' are identical. The Kullback-Leibler Divergence is *not* symmetric with respect to X and X'.

$D(X'||X)$ can be interpreted as a measure of the additional cost penalty (in bits, ▶ Entropy) to predict the outcome of X' if one assumes a model measure p instead of the true probability p'. When the Kullback-Leibler divergence becomes infinite, this corresponds to an outcome x' for X' with $p(x') > 0$ which "had not been foreseen" in X, i.e., where the model X assumes $p(x) = 0$.

An important special case of the Kullback-Leibler divergence is the ▶ mutual information between two variables X and Y; if their joint probability is $p(X, Y)$ and the respective individual probabilities $p(X)$ and $p(Y)$, one has

$$I(X,Y) = D(p(X,Y)||p(X)p(Y)) = \sum_{(x,y)\in\chi\times y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}.$$

Expressing mutual information as Kullback-Leibler divergence shows that the mutual information between two variables is same as the extra cost (in bits, ▶ Entropy) incurred when one models two random variables X and Y as independent variables when they are actually jointly dependent.

Note that the closely related ▶ (Shannon) Information compares *joint* probabilistic variables over possibly *unrelated* outcome sets, whereas the Kullback-Leibler divergence compares random variables over the *same* outcome set, but with possibly *unrelated* probabilistic variables.

References

Cover TM, Thomas JA (2006) Elements of information theory, 2nd edn. Wiley, Hoboken

Kullback–Leibler Divergence

▶ Information Gain

Kuramoto Model

Xiaojuan Sun
Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University of Beijing, Beijing, China

Definition

The Kuramoto model is a mathematical model first proposed by Yoshiki Kuramoto used to describe collective synchronization. It is a model for the behavior of a large set of coupled oscillators. Mathematical formulation of this model was motivated by the behavior of systems of chemical and biological oscillators, and can be applied to many other examples. The Kuramoto model shows that a population of coupled oscillators can spontaneously lock to common frequency, despite the inevitable differences in the natural frequencies of the individual oscillators (Strogatz 2000; Acebrón et al. 2005).

The Kuramoto model consists of N coupled phase oscillators whose phase is represented by $\theta_i(t)$. Each oscillator $\theta_i(t)$ has its own intrinsic natural frequency ω_i. The frequencies ω_i are assumed to

distribute with probability density $g(\omega)$. The Kuramoto model is given by

$$\dot{\theta}_i = \omega_i + \sum_{j=1}^{N} K_{ij} \sin(\theta_j - \theta_i), \quad i = 1, 2, \cdots, N. \quad (1)$$

The coupling matrix $K = (K_{ij})$ describes how the oscillators interconnect with each other. Many different models have been considered, such as nearest-neighbor coupling, hierarchical coupling, random long-range coupling, or even state-dependent interactions (Acebrón et al. 2005).

The mean-field coupling model, taking $K_{ij} = K/N$, was originally analyzed by Kuramoto. The mean-field model can be rewritten as

$$\dot{\theta}_i = \omega_i + \frac{K}{N} \sum_{j=1}^{N} \sin(\theta_j - \theta_i), \quad i = 1, 2, \cdots, N. \quad (2)$$

Here $K > 0$ is the coupling strength and the factor $1/N$ ensures that the model is well behaved as $N \to \infty$ (Strogatz 2000; Acebrón et al. 2005). Through the complex order parameter

$$r(t)e^{i\phi(t)} = \frac{1}{N} \sum_{j=1}^{N} e^{i\theta_j(t)},$$

we can rewrite Eq. 2 as

$$\dot{\theta}_i = \omega_i + Kr \sin(\phi - \theta_i), \quad i = 1, 2, \cdots, N. \quad (3)$$

Thus, each oscillator is coupled to the common average phase $\phi(t)$ with coupling strength given by Kr.

When the coupling strength $K = 0$, Eq. 3 yields $\theta_i = \omega_i t + \theta_i(0)$, which means that the oscillators rotate incoherently at their own frequencies. In the case of strong coupling with $K \to \infty$, we have $\sin(\phi - \theta_i) \to 0$, which implies $\theta_i \to \phi$, as $t \to \infty$. Thus, we have $r \to 1$ and the oscillators are completely synchronized. For intermediate coupling with $K_c < K < \infty$ ($K_c = 2/(\pi g(0))$ is the critical value), we can have partial synchronization with $0 < r < 1$ (Acebrón et al. 2005; Arenas et al. 2008).

References

Acebrón JA, Bonilla LL, Vicente CJP, Ritort F, Spigler R (2005) The Kuramoto model: a simple paradigm for synchronization phenomena. Rev Mod Phys 77:137–185

Arenas A, Díaz-Guilera A, Kurths J, Moreno Y, Zhou CS (2008) Synchronization in complex networks. Phys Rep 469:93–153

Strogatz HS (2000) From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators. Physica D 143:1–20

Kyoto Encyclopedia of Genes and Genomes Pathway

▶ KEGG Pathway Database