CaliBayes and BASIS: integrated tools for the calibration, simulation and storage of biological simulation models

Yuhui Chen, Conor Lawless, Colin S. Gillespie, Jake Wu, Richard J. Boys and Darren J. Wilkinson

Submitted: 2nd September 2009; Received (in revised form): 4th December 2009

Abstract

Dynamic simulation modelling of complex biological processes forms the backbone of systems biology. Discrete stochastic models are particularly appropriate for describing sub-cellular molecular interactions, especially when critical molecular species are thought to be present at low copy-numbers. For example, these stochastic effects play an important role in models of human ageing, where ageing results from the long-term accumulation of random damage at various biological scales. Unfortunately, realistic stochastic simulation of discrete biological processes is highly computationally intensive, requiring specialist hardware, and can benefit greatly from parallel and distributed approaches to computation and analysis. For these reasons, we have developed the BASIS system for the simulation and storage of stochastic SBML models together with associated simulation results. This system is exposed as a set of web services to allow users to incorporate its simulation tools into their workflows. Parameter inference for stochastic models is also difficult and computationally expensive. The CaliBayes system provides a set of web services (together with an R package for consuming these and formatting data) which addresses this problem for SBML models. It uses a sequential Bayesian MCMC method, which is powerful and flexible, providing very rich information. However this approach is exceptionally computationally intensive and requires the use of a carefully designed architecture. Again, these tools are exposed as web services to allow users to take advantage of this system. In this article, we describe these two systems and demonstrate their integrated use with an example workflow to estimate the parameters of a simple model of *Saccharomyces cerevisiae* growth on agar plates.

Keywords: Bayesian inference; distributed computing; SBML; stochastic models; web services

INTRODUCTION

As a result of advances in experimental techniques, biology has become a much more quantitative science. The capacity to answer questions ranging in scale from cell and molecular function through to population dynamics requires an increasing ability to acquire, store, and manipulate large volumes of raw data in a flexible, efficient manner. Moreover,
there is a growing realization that complex biological processes cannot be understood through the application of ever-more reductionist experimental programmes.

Mathematical modelling can provide key insights into the biological mechanisms underpinning much of the mass of biological data which is currently available. Indeed, there are distinct advantages of modelling a biological process with the rigour needed to build a mathematical model. For example, when constructing a model, gaps in current knowledge are highlighted [1, 2]. Even the very process of specifying a model can identify important quantities which remain unknown or unobserved. Also, qualitative verbal hypotheses are made quantitative, specific and conceptually rigorous [3, 4]. Further, models can yield quantitative as well as qualitative predictions [5, 6].

For all of these reasons, there has been a need to develop systems which aid the modelling and analysis components of the study of biological data. To this end, we have created a web service based modelling system known as the Biology of Ageing e-Science Integration and Simulation (BASIS) system [7]. The primary objective of BASIS is to help advance the understanding of the complex biology of ageing, where many different mechanisms act and interact at a range of different levels. Discrete stochastic simulation modelling is particularly relevant to biological investigation. For example, when considering sub-cellular models of biochemical processes where critical species have low copy-number, discrete simulation deals with the qualitative difference between the complete absence of a molecule from a system, and its presence at very low levels. This is unlike models based on the continuum approximation which often simply cannot deal with the extinction of a species [8]. Stochastic simulation is generally appropriate for describing biological systems as their intrinsic complexity often leads to small interactions (e.g. environmental interactions or cell-cell interactions) which are not included explicitly in a model. Thus stochastic models often provide an excellent framework for dealing with the combined effect of many unmodelled weak processes and interactions. Furthermore, these models encompass those needed to describe and design time course biological experiments in which it is not possible to have complete control over the initial conditions of the system (e.g. cell cycle synchrony or isogenicity across a population of cells). Stochastic models play a particularly important role in ageing research. In this area, ageing is typically described as the cumulative result of small amounts of random damage and the propagation of this damage throughout the lifetime of a cell or organism requires appropriate characterization of random effects, such as DNA damage, protein damage or the accumulation of DNA mutations which lead to tumourigenesis.

Although BASIS has been designed with ageing research in mind, its capabilities are generic to a wide range of other biological systems. Our system aims to make both existing and new models accessible to the research community in a way that facilitates users to adapt models and to run simulations themselves. It also makes publicly available a relatively complex, powerful and expensive computational architecture that is necessary for inferring parameters in biological stochastic models when using cohorts of simulation results. BASIS has adopted the Systems Biology Markup Language [9] (SBML) as a model description language. This is an XML-based computer-readable format for representing models of biochemical reaction networks.

The BASIS project is supported by a team from a wide range of disciplines including the biological sciences, mathematical and statistical sciences, and computer science. A key aim is to facilitate collaboration between experimental scientists and mathematical modellers; see [10] for an example. By sharing and integrating models and data, advances have already been made in our understanding of ageing. BASIS also provides open source downloadable tools in addition to a comprehensive online simulation system and modelling environment; see ref. [11] for details.

The CaliBayes project has a different focus and studies how to overcome the computational difficulties encountered when estimating parameter values in stochastic models. Currently, there are several computational tools available for estimating parameter values in deterministic models described in SBML, such as COPASI [12], BioBayes [13], SloppyCell [14] and GNUMCSim[15]. Note that developers have focussed on the SBML standard as it is by far the most widely used by researchers in the area. However, the tools listed above do not cater for (discrete) stochastic kinetic models. Some tools are available for parameter inference for statistical models (e.g. OpenBUGS [16]), but these tools cannot deal with the dynamic stochastic kinetic models used by
modellers to describe biological systems let alone models written in SBML. The aim of the CaliBayes project has been to fill this gap by providing inference tools aimed at stochastic SBML models. CaliBayes tools can also be used for the simpler but useful problem of parameter inference for deterministically modelled processes observed with error.

The method underlying the CaliBayes tools works essentially through a stochastic comparison of simulated model output obtained from different parameter values with what is often very noisy biological experimental data. The project uses Bayesian methods to provide posterior distributions for parameter values which describe uncertainty about their true values. These distributions provide a more natural representation of knowledge about parameters than do, for example, point estimates obtained from maximum likelihood or least-squares method. Also Bayesian methods are particularly suited to making inferences in complex stochastic biological models using partially observed time course data [17]. Much biological data are partially observed whether it consists of a continuous process measured at a few time points or whether key variables or components in the model are not observed at all. The modelling framework we use allows for the underlying biological model to be a deterministic model or a stochastic model. The observational (stochastic) error model describes the (random) discrepancies between model outputs and experimental data. Put together, these models describe a stochastic model for the experimental data and it is this overall stochastic model that is calibrated to the data. Another feature of CaliBayes is that it allows prior information about modelled parameters to be used to optimize inferences. This information can be obtained from, for example, the literature or the analysis of previous similar experiments using a simplified model structure and experimental measurement error. Additionally, distributional information can be included about the initial levels of model variables.

The CaliBayes project provides a complete suite of tools necessary for performing Bayesian parameter inference in stochastic biological models. These include: (i) an R package for formatting experimental data and the user’s prior beliefs about parameter values and initial conditions of model variables (typically initial species values or concentrations); (ii) web service tools for forward simulation of these models (deterministically, stochastically or by using hybrid methods) and (iii) tools for parameter inference. The CaliBayes R package (calibayesR) also provides an optional interface to the CaliBayes web services within the R computing environment. Novel methods for Bayesian inference in stochastic biological models have been developed and tested by our group [18–21]. The CaliBayes project exploits these techniques, together with significant associated computing power, and makes them available for public access via web services.

BASIS and CaliBayes are unusual in that they allow users to interact with these advanced modelling facilities through a web service API. Furthermore, parts of the BASIS system can be accessed through a user-friendly web interface (which utilizes the same web services), or downloaded and implemented locally.

The use of web service technology in biological modelling is gradually increasing along with the ambition of modellers to study ever larger and more complex systems, and in greater detail, using more accurate modelling techniques. Although such systems currently typically run on small clusters in academic institutions, looking ahead there is likely to be increasing use made of large GRIDs and cloud computing approaches. Software systems exploring the use of web service interfaces include popular simulation tools such as COPASI [12] and the Systems Biology Workbench [22]. This is in addition to the use of web-based modelling systems [23] that often exist primarily to provide cross-platform services that do not require downloading and installation of software.

The software described in this article is available from www.calibayes.ncl.ac.uk/Documentation and www.basis.ncl.ac.uk/software.

THE BASIS SYSTEM

BASIS is a system for model definition, simulation and visualization for stochastic models written in the SBML language. It is a collection of software tools running on a large cluster of CPUs and is exposed through several web services (Figure 1) which also drive a simple web-based GUI. The main advantage for users is that they gain access (via web-services) to a large and powerful computing cluster running parallel jobs of stochastic SBML simulators and to results from other models stored in a dedicated database. One tool we provide for using BASIS web services is a library of R functions (basisR) and these
functions access the web services directly. The web services interact with a PostgreSQL database [24], which stores user information, models and simulation results, and triggers simultaneous simulation jobs from different users. Simulation jobs are triggered via a Condor [25] job scheduler, which efficiently distributes parallel jobs (for cohorts of stochastic simulations from a single run from a single user) across a 96-CPU Beowulf cluster. All details of the underlying technology are hidden from the user.

To interact with the services that BASIS provides, a user must first register: this is simply to allow the user to retrieve their models and simulation results. A user can register either by visiting the web-site [26] or by using the web service createUser. When registering, a valid email address is required to discourage potential abuses of the system.

The majority of the web services provided by BASIS require a session ID as an argument. A session ID is obtained with the getSessionId web service. As each user logs on, a unique valid session ID is generated and returned to the user, with each being rendered invalid after several hours of inactivity.

Initially, when a user places an SBML model into the BASIS system, the model is designated private and is only accessible by that user. However, a user can make their model public (after publication, say) but once a model is made public, it can not be deleted.

Every model entered into the BASIS system is assigned a model Uniform Resource Name (URN) as a unique identifier. The model URN has the form urn:basis.net:model:#1 where #1 is an integer. A user can simulate from their model via the Gillespie algorithm by using a stochastic simulator (called Gillespie2), which is built using the efficient GNU scientific libraries and libSBML [27]. The simulator currently supports local and global parameters, events (without delays), assignment rules and randomly distributed parameters and species. It can be downloaded separately and installed on local machines if required [11]. One feature of BASIS is that when a model has been simulated on the system, the results are automatically associated with that particular model. Therefore, when a model is made public, all its associated simulation data also become public. This allows users to share their results with others and thereby reduce the overall computational

Figure 1: BASIS Architecture.
load on the system (stochastic simulations are generally much slower than deterministic ones. Further details of the BASIS web services can be found in ref. [28].

**THE CALIBAYES SYSTEM**

CaliBayes is a suite of tools for performing parameter inference in stochastic biological models specified in SBML using experimental data. Its architecture is shown in Figure 2. Parameter inference (or model calibration) for such models is typically performed by comparing discretely observed time course experimental data with simulated results from a mathematical dynamic simulation model describing the biological system of interest. Often these models are based around sets of coupled differential and algebraic equations (DAEs) and fitted by using least-squares. This process is equivalent to fitting the stochastic model whose mean is the described by the DAE and with independent and normally distributed errors. Typically, these methods are used only to provide point estimates of parameter values. CaliBayes, on the other hand, uses such simulators to perform rapid parameter inference on this type of model (and other inherently stochastic models) by using Bayesian sequential MCMC methods to obtain posterior distributions which describe uncertainty about model parameter values. The hardware and algorithms driving CaliBayes are made available via simple web services and an R library (calibayesR). CaliBayes has been designed to be completely modular, and can utilize any SBML-compliant simulation engine (deterministic or stochastic) via a transparent interface. It is unique in its ease of use and the richness of the information it provides of relevance to biological modelling.

Stochastic models are often preferred to deterministic models in a systems biology context for several reasons. For example, environmental interactions and initial system states can be imperfectly characterized or ignored in biological systems due to their complexity. Also, even when it is possible to replicate initial conditions exactly, repeated experiments often produce different outcomes. This inherent stochasticity can best be captured by using a stochastic model.
Discrete effects are particularly important in a systems biology context when modelling species with low copy numbers, for example, on the molecular scale. In this situation, the discreteness of the underlying biological process plays an important role in producing the experimental data. This effect is well known and is used in, for example, epidemiological models to capture the qualitative difference between the complete absence of a species (irreversible extinction) and its presence at a very low level (reversible decline) [8]. Discrete stochastic models are a natural way to describe the interaction of biochemical species [29], neatly capturing both stochasticity and discreteness. The main simulation engines used by CaliBayes are COPASI [12], FERN [30] and BASIS and all contain implementations of the discrete stochastic Gillespie algorithm. CaliBayes uses sequential Bayesian MCMC methods [19] which are ideally suited to stochastic kinetic models.

Posterior distributions describing uncertainty about parameter values are a natural output from the Bayesian methods used to calibrate our stochastic models. These form an ideal summary of the information about the parameters in the experimental data. The output also allows for the testing of hypotheses such as whether there is evidence for differences between parameter values in different experiments. This contrasts with the output of many other fitting procedures which only have point estimates of model parameters, as this severely restricts the role that modelling can have in the testing of scientific hypotheses. These posterior distributions for parameters also provide information on identifiability and confounding; see ref. [31] for further discussion.

Unlike the BASIS system, CaliBayes has no independent method for model storage (though this can be done via the BASIS storage system) or any long-term database storage for posterior distribution results. Instead, posterior results are stored only temporarily until they have been downloaded to the user’s local machine via the session ID used to generate them. The user then takes responsibility for their results and they are deleted from the CaliBayes system. The main reason for not providing long-term storage is that parameter inference is usually an iterative process with many intermediate steps assessing the quality of the inference, and so such storage would not be sufficiently beneficial for users given the increased complexity of the CaliBayes architecture.

The CaliBayes system is deployed, maintained and made publicly available on hardware based at Newcastle University, UK as a medium-powered example of its operation. However, all of its components are freely available for local deployment, and that is envisaged to be the primary mode for its use. Thus, CaliBayes can benefit from the availability of large amounts of computing power and scales well to take advantage of available hardware. Although CaliBayes uses cutting-edge Bayesian computationally intensive algorithms, most analyses of models of medium size (~8 species) using moderate (but not large) datasets (~20 time points per species) can be completed overnight.

CaliBayes software architecture
The CaliBayes software consists of a number of interacting service components. Each component may be deployed on a different machine or all components on the same machine. For the publicly available system, hosted at Newcastle, users can interact with these immediately by using web services or by downloading the CaliBayes API [32]. The main calibration services are as follows.

CaliBayes simulator interface
CaliBayes makes use of third-party SBML-compliant simulators for forward simulation. Such simulators may be either deterministic or stochastic, depending on the nature of the model to be calibrated. Any simulator can be used for this purpose so long as a SOAP [33] web services interface is also provided that conforms to the standard CaliBayes simulator interface. Example interfaces are given for COPASI (deterministic, stochastic and hybrid) and FERN (stochastic, SBML assignment rules not allowed) simulators in the publicly accessible demo system. The simulators are used for typically millions of short simulations per CaliBayes job, and so powerful CPUs are useful.

CaliBayes calibration engine
The main back-end computational service implementing the sequential Bayesian MCMC algorithm for model calibration. It is not intended that this service is accessed directly by users. This engine typically initializes millions of short simulation runs via the CaliBayes simulator interface, and therefore requires a wide bandwidth connection between this machine and those running the simulators.
CaliBayes data integrator
The main user-level calibration service. This service allows the calibration of a model based on multiple time series, each of which may consist of measurements of different species or other model components and at different time points.

In addition to the main CaliBayes software components, we have developed a support package (calibayesR) for the R statistical programming language [34]. This package has been designed to make it straightforward to generate, process and visualize the XML documents consumed and produced by the CaliBayes services. It also includes functions for accessing the CaliBayes web services directly from within the R environment. This allows users to take full advantage of the graphical and statistical capabilities of R (which is freely available on all platforms) by writing entire workflows within R, including data formatting, prior generation and posterior visualization. Note that this does not preclude accessing the web services via any other tool that the user finds more convenient (e.g. Python, Java or Taverna [35]).

A COMBINED BASIS AND CALIBAYES WORKFLOW
The CaliBayes system has been used to analyse a range of stochastic models in systems biology. For example, [36] use it to study a complex nonlinear continuous time latent stochastic process model describing the levels of two key proteins involved in the cellular response to DNA damage, captured at the level of individual cells in a human cancer cell line. In this section, for ease of exposition, we study a simpler model describing the logistic growth of *Saccharomyces cerevisiae* colonies spotted onto solid agar plates. Spot size is represented by species S, and K and r are parameters for the spot carrying capacity and growth rate, respectively. We also assume that the data are normally distributed about these model values with measurement error precision S.tau (variance = 1/S.tau).

The unknown quantities in the model are K, r, the initial spot size S(0) and the measurement error precision S.tau. These are calibrated to the data as follows. First, the data are split into sequential batches, each containing observations at b time points. Simulating values from the prior distribution for the unknown quantities and then forward simulating from the stochastic model gives a distribution of values for S at the each of the time points in the first batch. Comparing this distribution with the observed values in the experimental data gives us information on which values of the unknown quantities are more realistic (in a probabilistic sense). Continuing this process by first including data at the second batch and then the third (and so on) eventually gives us the posterior distributions of these quantities calibrated to the whole dataset. First the SBML-shorthand is converted to SBML by using our mod2sbml web service. Then the calibayesR package is loaded to enable seamless access to the CaliBayes web services from within the R environment.

Listing 1 describes the model in SBML-shorthand [11, 37]. This shorthand provides a very useful human readable and editable model representation and can reliably convert to and from full SBML.

**Listing 1: The Logistic Model**

```plaintext
@model: 2.1.2 = LogisticModel:YeastSpot.Growth
  # Model of growth in photographed area of S. Cerevisiae spotted onto agar.
  # Growth arises from the population dynamics of merging yeast colonies.
  @units
  substance = item
  @compartments
  tile = 1.0
  @parameters
  # Carrying capacity (pixels)
  K.
  # Rate parameter (per day)
  r.
  @species
  tile:S
  @reactions
  @r = Proliferation
  # Logistic growth is like autocatalysis
  S⇒S + S
  r*S*(1-S/K)
```

This model describes the logistic growth of *S. cerevisiae* (baker’s yeast) spots growing on solid agar plates. Spot size is represented by species S, and K and r are parameters for the spot carrying capacity and growth rate, respectively. We also assume that the data are normally distributed about these model values with measurement error precision S.tau (variance = 1/S.tau).

The unknown quantities in the model are K, r, the initial spot size S(0) and the measurement error precision S.tau. These are calibrated to the data as follows. First, the data are split into sequential batches, each containing observations at b time points. Simulating values from the prior distribution for the unknown quantities and then forward simulating from the stochastic model gives a distribution of values for S at the each of the time points in the first batch. Comparing this distribution with the observed values in the experimental data gives us information on which values of the unknown quantities are more realistic (in a probabilistic sense). Continuing this process by first including data at the second batch and then the third (and so on) eventually gives us the posterior distributions of these quantities calibrated to the whole dataset. First the SBML-shorthand is converted to SBML by using our mod2sbml web service. Then the calibayesR package is loaded to enable seamless access to the CaliBayes web services from within the R environment.
Figure 3: Combined CaliBayes and BASIS workflow.
We demonstrate the process by calibrating this model to the experimental dataset shown in Figure 4. These data are growth measures (spot areas) determined from photographic images of the plates and are included as a data frame in the calibayesR package. They can be accessed after loading the SBMLModels package and calling data (LogisticModel). The information describing prior uncertainty about the unknown quantities (shown in Figure 5) is represented by similar data frames containing n samples from the prior distribution. These data frames are converted to CaliBayes compliant XML strings by using the createCalibayes function. We now modify our prior distribution by incorporating information in the experimental data. Before proceeding, an XML string is needed to describe MCMC tuning parameters such as burn-in and thinning. Note that, as with all calibayesR functions which utilize CaliBayes web services, this function requires a working WSDL address describing the location of the local CaliBayes web services.

Now we are ready to start the CaliBayes engine. First, the SBML model and the tuning and prior objects are passed to the CaliBayes submit web service and this returns a session ID. This session ID is then used to repeatedly call the CaliBayes isCalibayesReady web service to check if the job is complete. Once this web service returns TRUE, we execute the getPosterior web service and receive an XML document containing values from the posterior distribution. Figure 6 displays this information and, for each quantity, shows the trace plot, the autocorrelation function and the (marginal) posterior distribution. When the MCMC algorithm has converged, as in this case, trace plots should look to be randomly traversing the posterior distribution along the iterations and autocorrelations should be low.

**Figure 4:** Three experimental time courses of the growth of genetically identical *S. cerevisiae* spots on solid agar plates.

**Figure 5:** Prior distribution of the spot carrying capacity K, the growth rate r, the initial spot size S(0) and the error precision S.tau.
Such plots are also indicative of there being no identifiability issues for parameters within the model. Note that the density plots have much smaller variability than those in Figure 5. This is due to many values in the prior distribution being (probabilistically) inconsistent with the experimental data. The posterior output can be used to construct rough estimates and/or plausible ranges for these unknown quantities. For example, the plots show that $K$ has a value around 9500, $r$ around 3.2 and $S.tau$ around $1.5 \times 10^{-5}$.

We can also use the output of the CaliBayes software to conduct in silico experiments and, for example, determine the posterior predictive distribution of an experimental time course. This is the distribution of the time course allowing for posterior uncertainty in the model quantities, the inherent stochasticity in the biological model and the measurement error. It can be obtained by submitting the model, the posterior distribution for the quantities $K$, $r$ and $S.tau$, and the prior distribution for the initial species level $S(0)$ and making repeated calls to the BASIS forwardSimulate web service (as shown in the workflow given in Figure 3). The posterior predictive distribution for this model and data is given in Figure 7. It shows that the model fits reasonably well to these data.

The results of this analysis, in the form of the model and fixed values for the unknown quantities (set at their median values in the posterior distribution) have been deposited on the BASIS website and are available for use in the research community. These results were archived via the basisR package by using the BASIS getSessionID web service and then the returned valid session ID to submit the SBML model via the putSBML web service.
CONCLUSION
Stochastic simulation of biological processes is highly desirable as it can capture significant intrinsic, unmodelled variation and environmental interactions in complex biological systems. Also, discrete simulation models are particularly useful for modelling biochemical reactions in which species copy numbers are low since discrete effects can be important at this low level. Producing large cohorts of simulation results for discrete stochastic biochemical models is generally time consuming, particularly on small numbers of CPUs (a typical workstation for example), rendering this technique slow or even impractical for parameter inference. BASIS and CaliBayes are integrated systems for the rapid simulation of cohorts of discrete stochastic realizations from SBML models, and for parameter inference for these models based on Bayesian sequential MCMC algorithms, each deployed on a carefully constructed computational architecture. CaliBayes can also deal with continuous DAE models and provide posterior distributions for parameters assuming, for example, normal errors. These systems have been designed to achieve a workload throughput which is sufficiently high to provide practical and viable tools for discrete stochastic simulation and Bayesian inference for parameters in these models. This is achieved by utilizing a dedicated computer cluster running simulation engines, supported by scheduling software, inference algorithms, databases and high-speed network connections, all exposed to the public via web services. This complex, technical environment is also made available to users (through the same web services) via simple and easy-to-use R client libraries (calibayesR and basisR), and these libraries allow straightforward construction of prior distribution and plotting of posteriors and simulated results. In this article, we have demonstrated how to use these packages to access CaliBayes web services with a combined workflow for inferring logistic equation parameter values from experimental data describing the growth of *S. cerevisiae* cultures on solid agar.

The BASIS and CaliBayes systems are computationally intensive. Currently, at Newcastle, BASIS is running on a cluster of 96 CPUs and CaliBayes is running on 32 CPUs. Both of these systems can be scaled to service more users or perform more simulations (thereby reducing queuing time for simulation jobs), and they scale linearly with available computing power. All software components are freely available for local deployment, but a coordinated strategy for sharing resources between academic institutions, for example, by distributing CaliBayes and BASIS jobs across hardware on different sites worldwide, would best be achieved using GRID technology. Another alternative strategy, which would move responsibility for hardware maintenance away from academic institutions and improve reliability of service, would be to utilize commercially available ‘Cloud’ technologies such as Amazon’s EC2. Given the limited financial resources of individual academic institutions and lack of long-term funding for research projects, a Cloud-computing solution currently seems to be the most viable way to achieve ever higher levels of throughput from services such as CaliBayes and BASIS. However, academic funding models need to evolve to allow computing resources to be considered as a service rather than a capital investment before such an approach is likely to gain widespread adoption.

Key Points
- Stochastic simulation is computationally intensive, but necessary for understanding complex biological processes
- Bayesian parameter inference for complex models using experimental data is exceptionally computationally intensive
- Both stochastic simulation and Bayesian parameter inference are amenable to parallel and distributed computing approaches: we present BASIS and CaliBayes as examples, both of which exploit web-services and use a service-oriented architecture.
- The CaliBayes project provides the only tools currently available for analysing stochastic SBML models.
FUNDING
This work was supported by the Biotechnology and Biological Sciences Research Council [BEP17 042, BBSC16 550, BBC0082 001] with contributions from the Engineering and Physical Sciences Research Council, Medical Research Council, Department for Trade and Industry and Unilever Corporate Research.

References