Bayes linear graphical models and computer simulators for complex physical system

Michael Goldstein
Dept. Mathematical Sciences,
Durham University *

* Graphical design: Jonathan Cumming

The state of the art in climate modelling

Large climate models take months to run on supercomputers. One of the biggest computers in the world is the Earth Simulator in Japan, which is often used for running climate models.

Leading climate models

One leading climate model at the moment is based at the UK Met Office. The climate model (HadSM3) has about 100 uncertain parameters, including:

1. *Large scale cloud.* Six parameters
2. *Convection.* Six parameters
3. *Sea ice.* Two parameters
4. *Radiation.* Four parameters
5. *Dynamics.* Four parameters
6. *Land surface.* Four parameters
7. *Boundary layer.* Four parameters

Leading climate models

One leading climate model at the moment is based at the UK Met Office. The climate model (HadSM3) has about 100 uncertain parameters, including:

1. *Large scale cloud.* Six parameters
2. *Convection.* Six parameters
3. *Sea ice.* Two parameters
4. *Radiation.* Four parameters
5. *Dynamics.* Four parameters
6. *Land surface.* Four parameters
7. *Boundary layer.* Four parameters

We have a few hundred evaluations of HadSM3, made over about three years. These evaluations are a central resource for the UK Climate Impacts Programme 2009 (UKCIP09), intended as a fairly definitive view about how climate change will impact the UK, including climate uncertainty statements.

Relating a model with a physical system

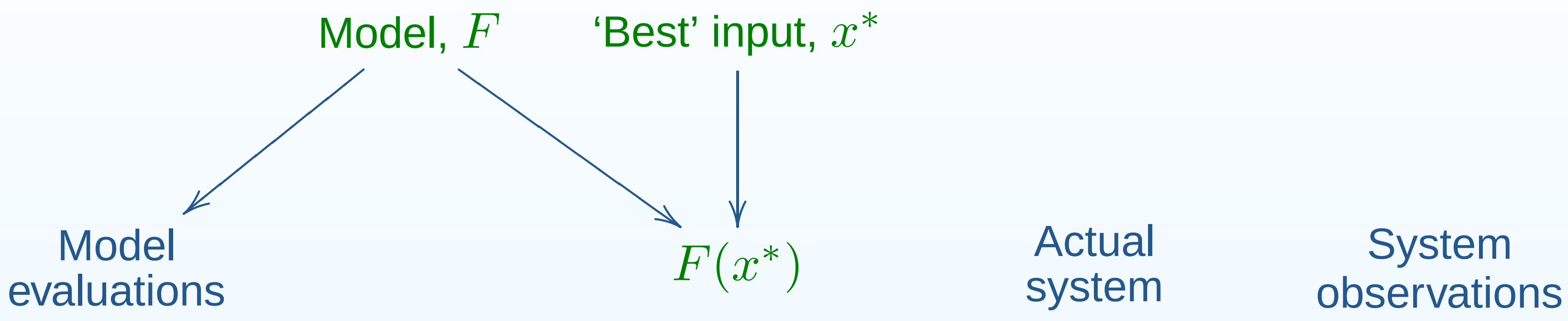
Model evaluations

Actual system

System observations

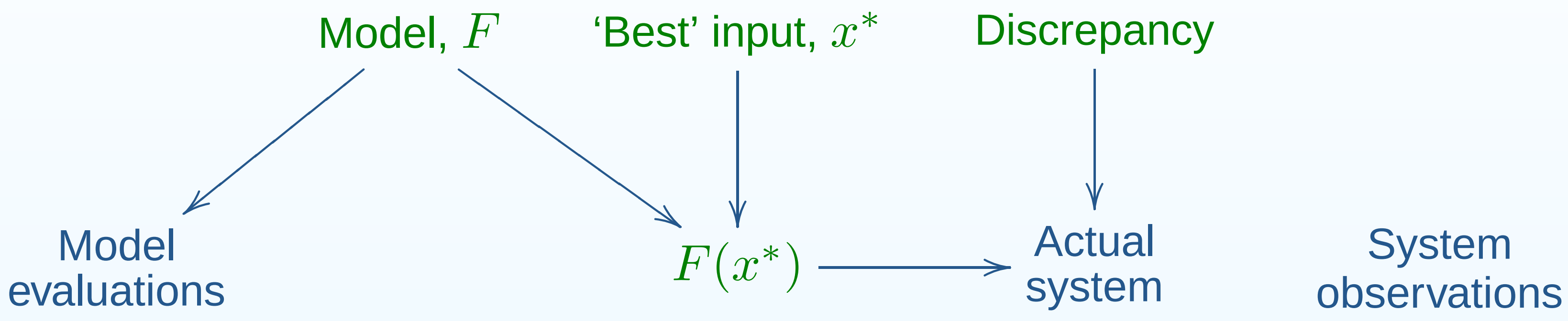
1. We start with a collection of model evaluations, and some system observations
2. We link the evaluations to the notion of a 'best' evaluation
3. We link the 'best' evaluation to the actual system
4. We incorporate measurement error into the observations
5. Our aim is to develop a unified Bayesian treatment of all these sources of uncertainty, within a natural graphical framework.

Relating a model with a physical system



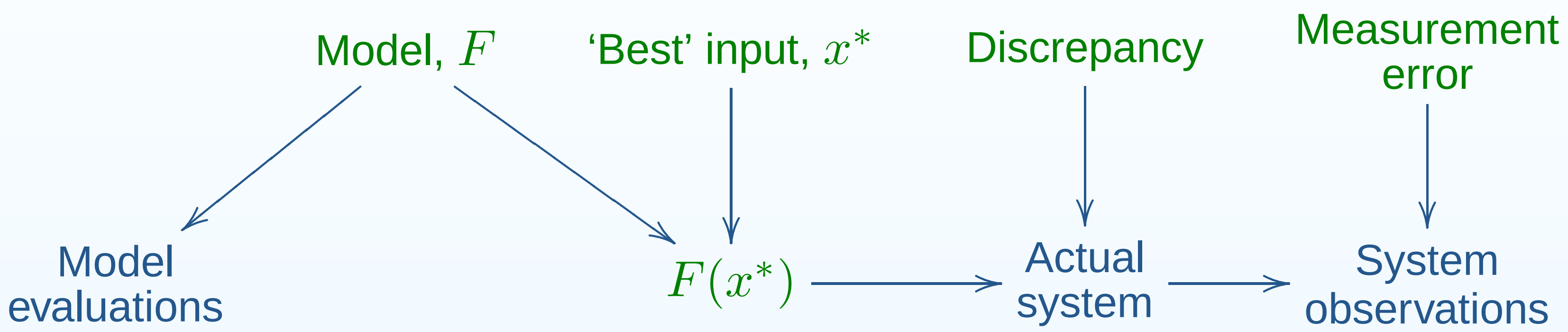
1. We start with a collection of model evaluations, and some system observations
2. We link the evaluations to the notion of a 'best' evaluation
3. We link the 'best' evaluation to the actual system
4. We incorporate measurement error into the observations
5. Our aim is to develop a unified Bayesian treatment of all these sources of uncertainty, within a natural graphical framework.

Relating a model with a physical system



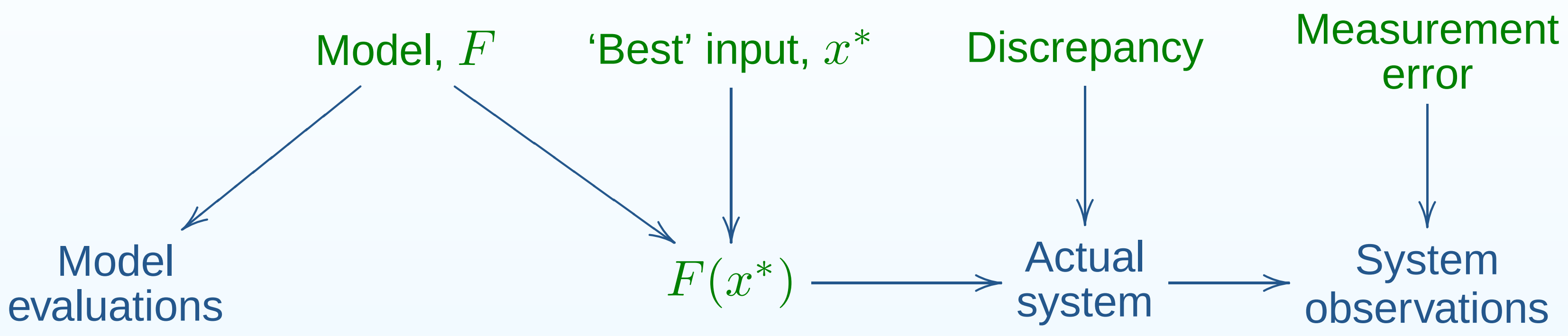
1. We start with a collection of model evaluations, and some system observations
2. We link the evaluations to the notion of a 'best' evaluation
3. We link the 'best' evaluation to the actual system
4. We incorporate measurement error into the observations
5. Our aim is to develop a unified Bayesian treatment of all these sources of uncertainty, within a natural graphical framework.

Relating a model with a physical system



1. We start with a collection of model evaluations, and some system observations
2. We link the evaluations to the notion of a 'best' evaluation
3. We link the 'best' evaluation to the actual system
4. We incorporate measurement error into the observations
5. Our aim is to develop a unified Bayesian treatment of all these sources of uncertainty, within a natural graphical framework.

Relating a model with a physical system



1. We start with a collection of model evaluations, and some system observations
2. We link the evaluations to the notion of a 'best' evaluation
3. We link the 'best' evaluation to the actual system
4. We incorporate measurement error into the observations
5. Our aim is to develop a unified Bayesian treatment of all these sources of uncertainty, within a natural graphical framework.

Representing beliefs about F using emulators

An *emulator* is a probabilistic belief specification for a deterministic function. Our emulator for component i of F might be

$$f_i(x) = \sum_j \beta_{ij}\, g_{ij}(x) + u_i(x)$$

where $B = \{\beta_{ij}\}$ are unknown scalars, g_{ij} are known deterministic functions of x, and $u(x)$ is a weakly stationary stochastic process.

Representing beliefs about F using emulators

An *emulator* is a probabilistic belief specification for a deterministic function.
Our emulator for component i of F might be

$$f_i(x) = \sum_j \beta_{ij}\, g_{ij}(x) + u_i(x)$$

where $B = \{\beta_{ij}\}$ are unknown scalars, g_{ij} are known deterministic functions of x, and $u(x)$ is a weakly stationary stochastic process.
We fit the emulator, $f = Bg(x) + u(x)$, given a collection of model evaluations, using our favourite statistical tools - generalised least squares, maximum likelihood, Bayes - with a generous helping of expert judgement.

Representing beliefs about F using emulators

An *emulator* is a probabilistic belief specification for a deterministic function.
Our emulator for component i of F might be

$$f_i(x) = \sum_j \beta_{ij}\, g_{ij}(x) + u_i(x)$$

where $B = \{\beta_{ij}\}$ are unknown scalars, g_{ij} are known deterministic functions of x, and $u(x)$ is a weakly stationary stochastic process.
We fit the emulator, $f = Bg(x) + u(x)$, given a collection of model evaluations, using our favourite statistical tools - generalised least squares, maximum likelihood, Bayes - with a generous helping of expert judgement.
$Bg(x)$ represents **global** variation and $u(x)$ represents **local** variation in F

Representing beliefs about F using emulators

An *emulator* is a probabilistic belief specification for a deterministic function. Our emulator for component i of F might be

$$f_i(x) = \sum_j \beta_{ij}\, g_{ij}(x) + u_i(x)$$

where $B = \{\beta_{ij}\}$ are unknown scalars, g_{ij} are known deterministic functions of x, and $u(x)$ is a weakly stationary stochastic process.

We fit the emulator, $f = Bg(x) + u(x)$, given a collection of model evaluations, using our favourite statistical tools - generalised least squares, maximum likelihood, Bayes - with a generous helping of expert judgement.

$Bg(x)$ represents **global** variation and $u(x)$ represents **local** variation in F

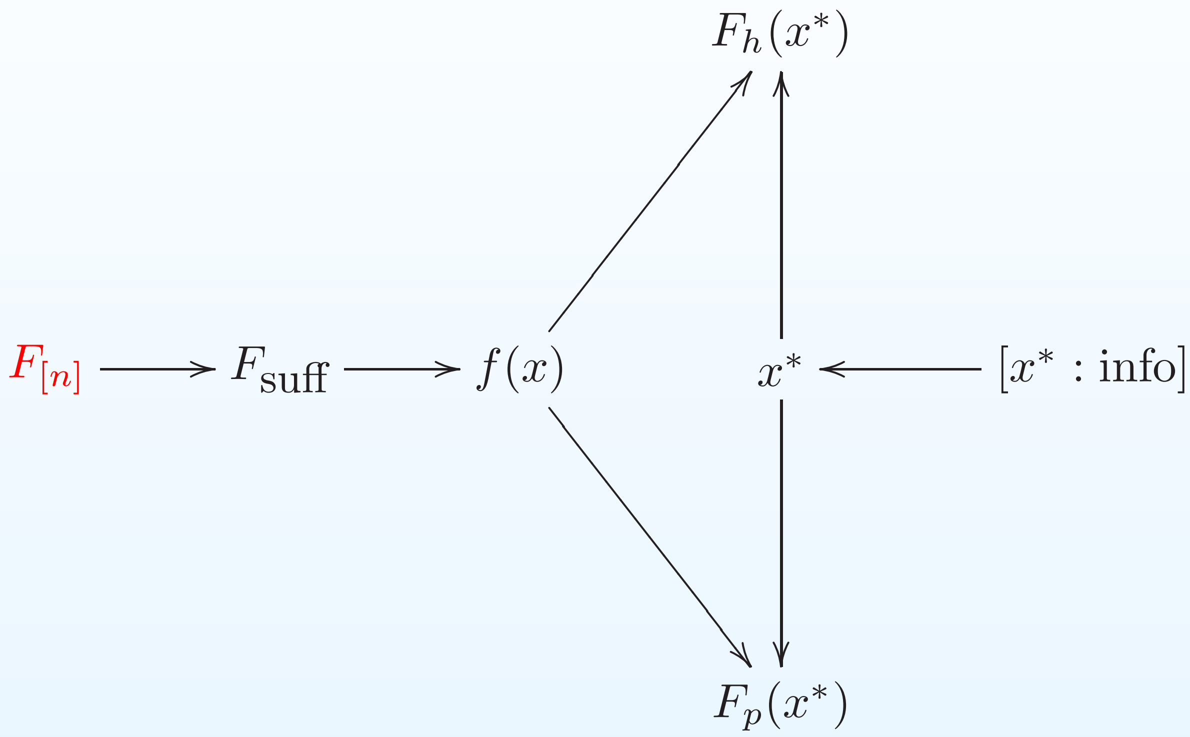
When the input dimension is high, relative to the number of function evaluations we can make, then most of what we may learn about the function comes through the global component. For simplicity, we therefore often suppose that our simulator judgements can be summarised by the global behaviour (as we don't learn much about local behaviour).

Function evaluations and emulator

$$F_{[n]} \longrightarrow F_{\text{suff}} \longrightarrow f(x)$$

$F_{[n]} = (F(x_1), F(x_2), \ldots)$: evaluations of F at inputs $x_1, x_2, \ldots$
F_{suff}: the global information from $F_{[n]}$ which forms emulator $f(x)$

Emulator and best evaluation

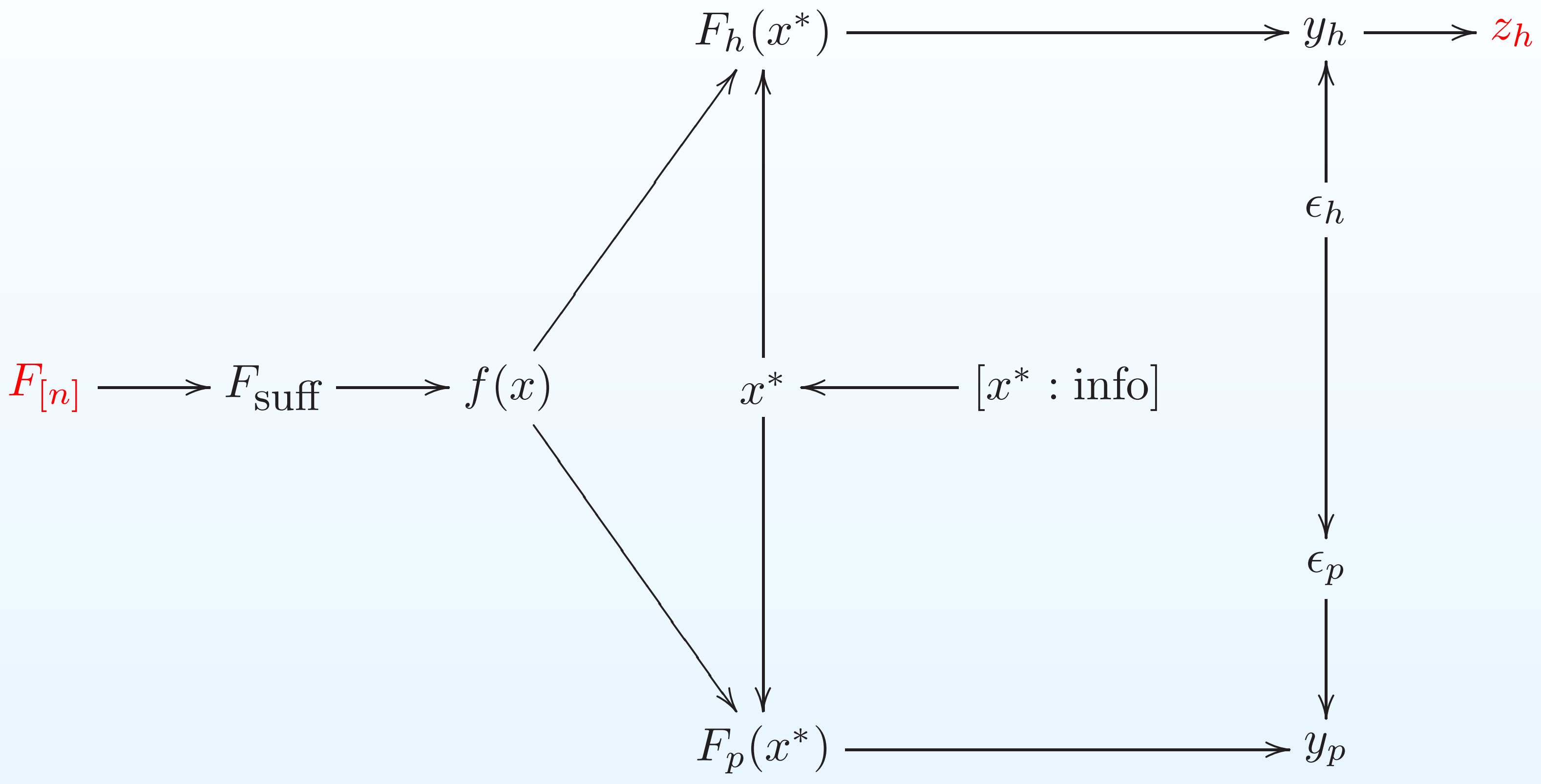


True system properties x^* with emulator $f(x)$ influence beliefs for

$F_h(x^*)$: components of F corresponding to **h**istorical outputs of F

$F_p(x^*)$: components of F corresponding to outputs of F to **p**redict

Best evaluation and system



$F_h(x^*)$ is informative for historical system values y_h observed with error as z_h
$F_p(x^*)$ is informative for system values y_p to predict.
ϵ_h, ϵ_p: the corresponding discrepancy terms between model and system

Bayes linear approach

For large scale problems a full Bayes analysis is very hard because
(i) it is difficult to give a meaningful full prior probability specification over high dimensional spaces;
(ii) the computations, for learning from data (observations and computer runs) and choosing informative runs, may be technically difficult.

Bayes linear approach

For large scale problems a full Bayes analysis is very hard because
(i) it is difficult to give a meaningful full prior probability specification over high dimensional spaces;
(ii) the computations, for learning from data (observations and computer runs) and choosing informative runs, may be technically difficult.
The Bayes Linear approach is (relatively) simple in terms of belief specification and analysis, as it is based only on the mean, variance and covariance specification which, following de Finetti, we take as primitive; see
Goldstein and Wooff (2007) Bayes Linear Statistics: Wiley.

Bayes linear approach

For large scale problems a full Bayes analysis is very hard because
(i) it is difficult to give a meaningful full prior probability specification over high dimensional spaces;
(ii) the computations, for learning from data (observations and computer runs) and choosing informative runs, may be technically difficult.
The Bayes Linear approach is (relatively) simple in terms of belief specification and analysis, as it is based only on the mean, variance and covariance specification which, following de Finetti, we take as primitive; see
Goldstein and Wooff (2007) Bayes Linear Statistics: Wiley.
$\mathsf{E}_z[y]$, $\mathsf{Var}_z[y]$, the expectation and variance for y adjusted by z, are given by

$$\mathsf{E}_z[y] = \mathsf{E}[y] + \mathrm{Cov}(y, z)\mathrm{Var}(z)^{-1}(z - \mathsf{E}[z]),$$
$$\mathsf{Var}_z[y] = \mathrm{Var}(y) - \mathrm{Cov}(y, z)\mathrm{Var}(z)^{-1}\mathrm{Cov}(z, y)$$

Bayes linear approach

For large scale problems a full Bayes analysis is very hard because
(i) it is difficult to give a meaningful full prior probability specification over high dimensional spaces;
(ii) the computations, for learning from data (observations and computer runs) and choosing informative runs, may be technically difficult.
The Bayes Linear approach is (relatively) simple in terms of belief specification and analysis, as it is based only on the mean, variance and covariance specification which, following de Finetti, we take as primitive; see
Goldstein and Wooff (2007) Bayes Linear Statistics: Wiley.
$\mathsf{E}_z[y]$, $\mathsf{Var}_z[y]$, the expectation and variance for y adjusted by z, are given by

$$\mathsf{E}_z[y] = \mathsf{E}[y] + \mathrm{Cov}(y, z)\mathrm{Var}(z)^{-1}(z - \mathsf{E}[z]),$$
$$\mathsf{Var}_z[y] = \mathrm{Var}(y) - \mathrm{Cov}(y, z)\mathrm{Var}(z)^{-1}\mathrm{Cov}(z, y)$$

Bayes linear analysis may be viewed as the appropriate analysis given a partial specification based on expectation.

Geometric description

For any collection $\boldsymbol{C} = (C_1, C_2, ...)$ of random quantities, we denote by $\langle \boldsymbol{C} \rangle$ the collection of (finite) linear combinations $\sum_i r_i C_i$ of the elements of $\boldsymbol{C}$. We view $\langle \boldsymbol{C} \rangle$ as a vector space.

Geometric description

For any collection $\boldsymbol{C} = (C_1, C_2, ...)$ of random quantities, we denote by $\langle \boldsymbol{C} \rangle$ the collection of (finite) linear combinations $\sum_i r_i C_i$ of the elements of $\boldsymbol{C}$. We view $\langle \boldsymbol{C} \rangle$ as a vector space.

Prior covariance is an inner product on $\langle \boldsymbol{C} \rangle$. If $\boldsymbol{C}$ is the union of the elements of the vectors $\boldsymbol{B}$ and $\boldsymbol{D},$ then the adjusted expectation of $Y \in \langle \boldsymbol{B} \rangle$ given $\boldsymbol{D}$, $\mathrm{E}_{\boldsymbol{D}}(X)$, is the orthogonal projection of Y into the linear subspace $\langle \boldsymbol{D} \rangle$, and adjusted variance, $\mathrm{Var}_{\boldsymbol{D}}(X)$, is the squared distance between Y and $\langle \boldsymbol{D} \rangle$.

Geometric description

For any collection $\boldsymbol{C} = (C_1, C_2, ...)$ of random quantities, we denote by $\langle \boldsymbol{C} \rangle$ the collection of (finite) linear combinations $\sum_i r_i C_i$ of the elements of $\boldsymbol{C}$. We view $\langle \boldsymbol{C} \rangle$ as a vector space.

Prior covariance is an inner product on $\langle \boldsymbol{C} \rangle$. If $\boldsymbol{C}$ is the union of the elements of the vectors $\boldsymbol{B}$ and $\boldsymbol{D}$, then the adjusted expectation of $Y \in \langle \boldsymbol{B} \rangle$ given $\boldsymbol{D}$, $\mathrm{E}_{\boldsymbol{D}}(X)$, is the orthogonal projection of Y into the linear subspace $\langle \boldsymbol{D} \rangle$, and adjusted variance, $\mathrm{Var}_{\boldsymbol{D}}(X)$, is the squared distance between Y and $\langle \boldsymbol{D} \rangle$.

In the usual Bayes formalism, $\langle C \rangle$ is the collection of all random variables defined over the outcome space. The inner product space is the Hilbert space of square integrable functions over the outcome space, with respect to the prior measure over the outcomes, with the covariance inner product.

Geometric description

For any collection $\boldsymbol{C} = (C_1, C_2, ...)$ of random quantities, we denote by $\langle \boldsymbol{C} \rangle$ the collection of (finite) linear combinations $\sum_i r_i C_i$ of the elements of $\boldsymbol{C}$. We view $\langle \boldsymbol{C} \rangle$ as a vector space.

Prior covariance is an inner product on $\langle \boldsymbol{C} \rangle$. If $\boldsymbol{C}$ is the union of the elements of the vectors $\boldsymbol{B}$ and $\boldsymbol{D}$, then the adjusted expectation of $Y \in \langle \boldsymbol{B} \rangle$ given $\boldsymbol{D}$, $\mathrm{E}_{\boldsymbol{D}}(X)$, is the orthogonal projection of Y into the linear subspace $\langle \boldsymbol{D} \rangle$, and adjusted variance, $\mathrm{Var}_{\boldsymbol{D}}(X)$, is the squared distance between Y and $\langle \boldsymbol{D} \rangle$.

In the usual Bayes formalism, $\langle C \rangle$ is the collection of all random variables defined over the outcome space. The inner product space is the Hilbert space of square integrable functions over the outcome space, with respect to the prior measure over the outcomes, with the covariance inner product.

Conditional expectation, given a sample, corresponds to orthogonal projection into the subspace of all functions over the sample space.

Geometric description

For any collection $\boldsymbol{C} = (C_1, C_2, ...)$ of random quantities, we denote by $\langle \boldsymbol{C} \rangle$ the collection of (finite) linear combinations $\sum_i r_i C_i$ of the elements of $\boldsymbol{C}$. We view $\langle \boldsymbol{C} \rangle$ as a vector space.

Prior covariance is an inner product on $\langle \boldsymbol{C} \rangle$. If $\boldsymbol{C}$ is the union of the elements of the vectors $\boldsymbol{B}$ and $\boldsymbol{D}$, then the adjusted expectation of $Y \in \langle \boldsymbol{B} \rangle$ given $\boldsymbol{D}$, $\mathrm{E}_{\boldsymbol{D}}(X)$, is the orthogonal projection of Y into the linear subspace $\langle \boldsymbol{D} \rangle$, and adjusted variance, $\mathrm{Var}_{\boldsymbol{D}}(X)$, is the squared distance between Y and $\langle \boldsymbol{D} \rangle$.

In the usual Bayes formalism, $\langle C \rangle$ is the collection of all random variables defined over the outcome space. The inner product space is the Hilbert space of square integrable functions over the outcome space, with respect to the prior measure over the outcomes, with the covariance inner product.

Conditional expectation, given a sample, corresponds to orthogonal projection into the subspace of all functions over the sample space.

Bayes linear analysis allows us to restrict prior specification and subsequent projection into the largest subspace of this full space that we are able to specify prior beliefs over.

Bayes linear belief nets

A, B, C are collections of random quantities.

Bayes linear belief nets

A, B, C are collections of random quantities.

C *separates* A and B, denoted $\lfloor A \perp\!\!\!\perp B \rfloor / C$, if

$$\mathrm{E}_{C \cup B}(A) = \mathrm{E}_C(A)$$

Bayes linear belief nets

A, B, C are collections of random quantities.
C *separates* A and B, denoted $\lfloor A \perp\!\!\!\perp B \rfloor / C$, if

$$\mathrm{E}_{C \cup B}(A) = \mathrm{E}_C(A)$$

Geometrically, $\lfloor A \perp\!\!\!\perp B \rfloor / C$ if the orthogonal complements of A and B in C,

$$[A - \mathrm{E}_C(A)], [B - \mathrm{E}_C(B)]$$

are orthogonal.

Bayes linear belief nets

A, B, C are collections of random quantities.
C *separates* A and B, denoted $\lfloor A \perp\!\!\!\perp B \rfloor \, / \, C$, if

$$\mathrm{E}_{C \cup B}(A) = \mathrm{E}_C(A)$$

Geometrically, $\lfloor A \perp\!\!\!\perp B \rfloor \, / \, C$ if the orthogonal complements of A and B in C,

$$[A - \mathrm{E}_C(A)], [B - \mathrm{E}_C(B)]$$

are orthogonal.
$\lfloor A \perp\!\!\!\perp B \rfloor \, / \, C$ is a generalised conditional independence property.

Bayes linear belief nets

A, B, C are collections of random quantities.
C *separates* A and B, denoted $\lfloor A \perp\!\!\!\perp B \rfloor \,/\, C$, if

$$\mathrm{E}_{C \cup B}(A) = \mathrm{E}_C(A)$$

Geometrically, $\lfloor A \perp\!\!\!\perp B \rfloor \,/\, C$ if the orthogonal complements of A and B in C,

$$[A - \mathrm{E}_C(A)], [B - \mathrm{E}_C(B)]$$

are orthogonal.
$\lfloor A \perp\!\!\!\perp B \rfloor \,/\, C$ is a generalised conditional independence property.
Therefore, graphical models expressing such belief separations (geometrically - the orthogonalities between subspaces) will have many of the same formal properties as do probabilistic graphical models.
Bayes linear graphical models have a close relationship with Gaussian graphical models.

Calibration via history matching

History Matching is concerned with learning about best inputs, x^*, using simulator evaluations and data, z. Using the emulator we obtain, for each input choice x, the adjusted values of $\mathrm{E}(f(x))$ and $\mathrm{Var}(f(x))$. We rule out regions of x space for which $F(x)$ is judged to be a very poor match to observed z.

Calibration via history matching

History Matching is concerned with learning about best inputs, x^*, using simulator evaluations and data, z. Using the emulator we obtain, for each input choice x, the adjusted values of $\mathrm{E}(f(x))$ and $\mathrm{Var}(f(x))$. We rule out regions of x space for which $F(x)$ is judged to be a very poor match to observed z.

To achieve this, we calculate, for each output $F_i(x)$, the implausibility:

$$I_{(i)}(x) = |z_i - \mathrm{E}(f_i(x))|^2 / \mathrm{Var}(z_i - \mathrm{E}(f_i(x)))$$

Calibration via history matching

History Matching is concerned with learning about best inputs, x^*, using simulator evaluations and data, z. Using the emulator we obtain, for each input choice x, the adjusted values of $\mathrm{E}(f(x))$ and $\mathrm{Var}(f(x))$. We rule out regions of x space for which $F(x)$ is judged to be a very poor match to observed z.
To achieve this, we calculate, for each output $F_i(x)$, the implausibility:

$$I_{(i)}(x) = |z_i - \mathrm{E}(f_i(x))|^2 / \mathrm{Var}(z_i - \mathrm{E}(f_i(x)))$$

This calculation can be performed univariately, or over sub-vectors. The implausibilities are then combined, such as by using $I_M(x) = \max_i I_{(i)}(x)$, identifying regions of x with large $I_M(x)$ as unlikely to be good choices for x^*.

Calibration via history matching

History Matching is concerned with learning about best inputs, x^*, using simulator evaluations and data, z. Using the emulator we obtain, for each input choice x, the adjusted values of $\mathrm{E}(f(x))$ and $\mathrm{Var}(f(x))$. We rule out regions of x space for which $F(x)$ is judged to be a very poor match to observed z.

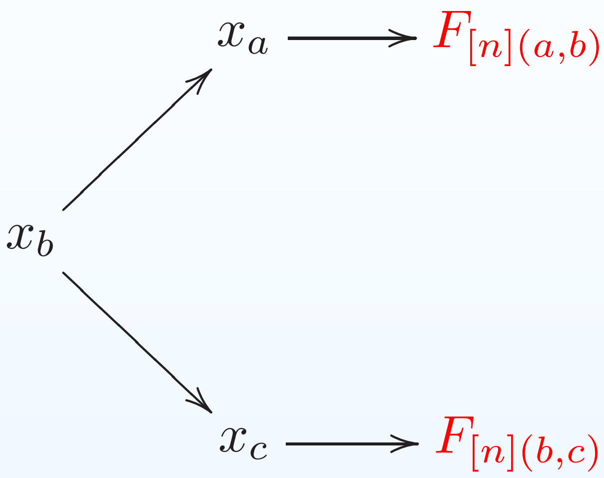
To achieve this, we calculate, for each output $F_i(x)$, the implausibility:

$$I_{(i)}(x) = |z_i - \mathrm{E}(f_i(x))|^2 / \mathrm{Var}(z_i - \mathrm{E}(f_i(x)))$$

This calculation can be performed univariately, or over sub-vectors. The implausibilities are then combined, such as by using $I_M(x) = \max_i I_{(i)}(x)$, identifying regions of x with large $I_M(x)$ as unlikely to be good choices for x^*.

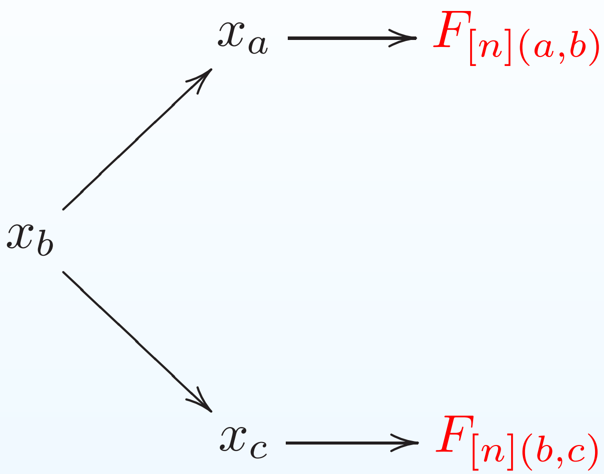
We iteratively refocus on the 'non-implausible' regions of the input space, by further model runs and refitting our emulator over the sub-region and repeating the analysis. This process is a form of iterative global search.

Causal structure and design



Functional graphical models are causal models on the functional inputs.

Causal structure and design

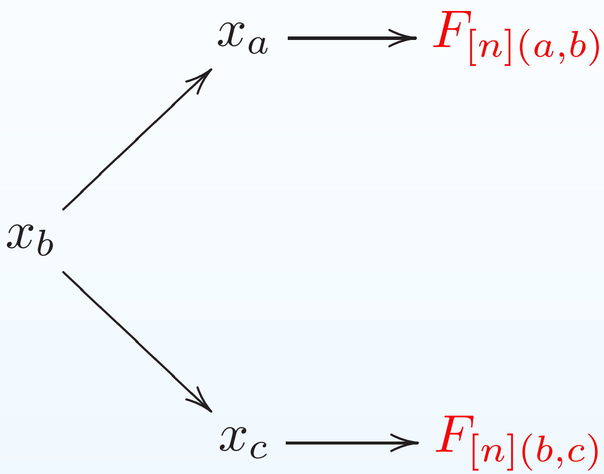


Functional graphical models are causal models on the functional inputs.

Here the outputs divide into three sets x_a, x_b, x_c.

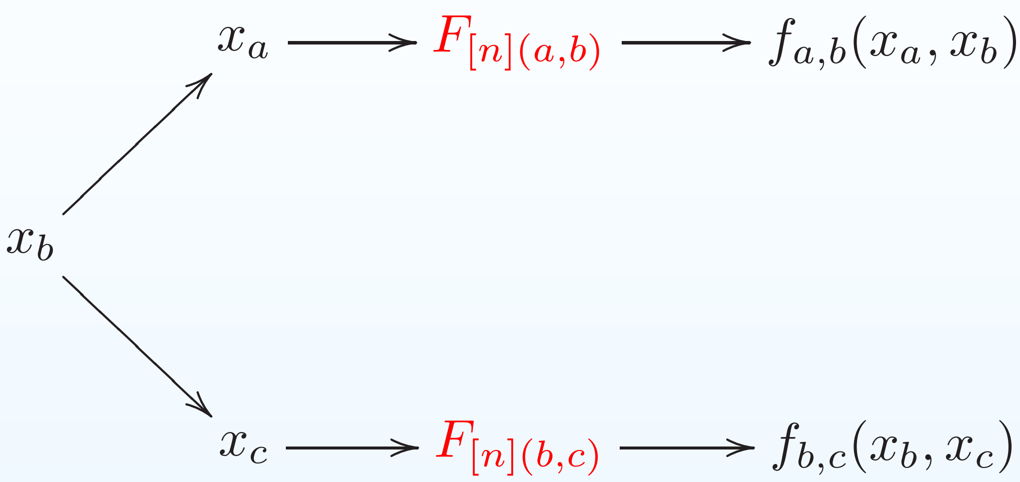
Outputs $F_{(a,b)}$, depend only on x_a, x_b. Outputs $F_{(b,c)}$, depend only on x_b, x_c

Causal structure and design



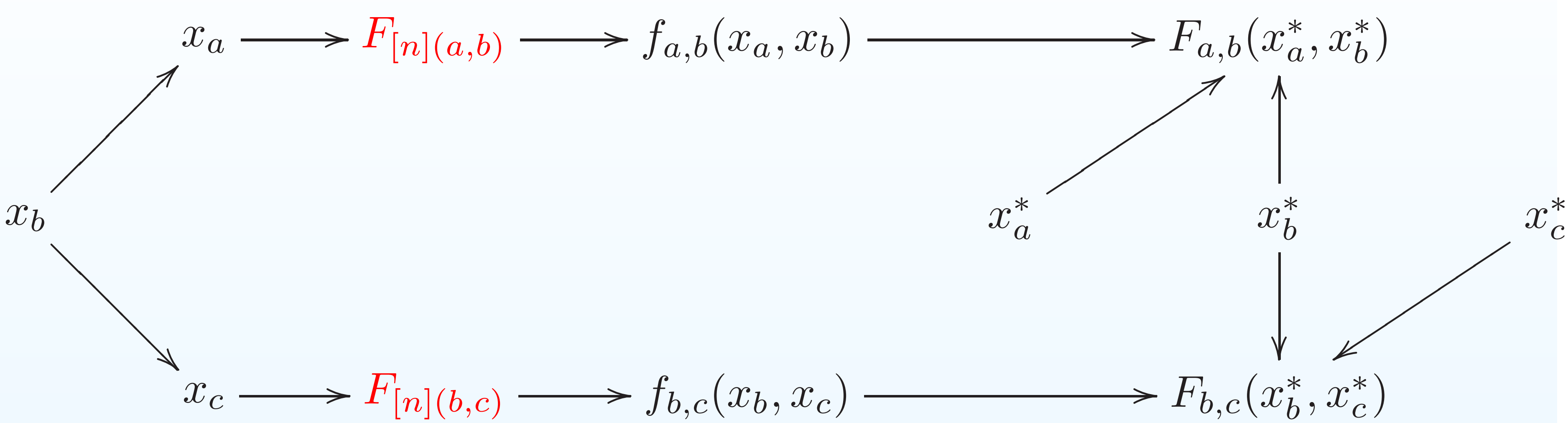
Functional graphical models are causal models on the functional inputs.
Here the outputs divide into three sets x_a, x_b, x_c.
Outputs $F_{(a,b)}$, depend only on x_a, x_b. Outputs $F_{(b,c)}$, depend only on x_b, x_c
Therefore, we can design a collection of n evaluations, $F_{[n](a,b)}$ and $F_{[n](b,c)}$ independently given our design for x_b (which is enormously helpful in reducing dimensionality)

Design and emulation



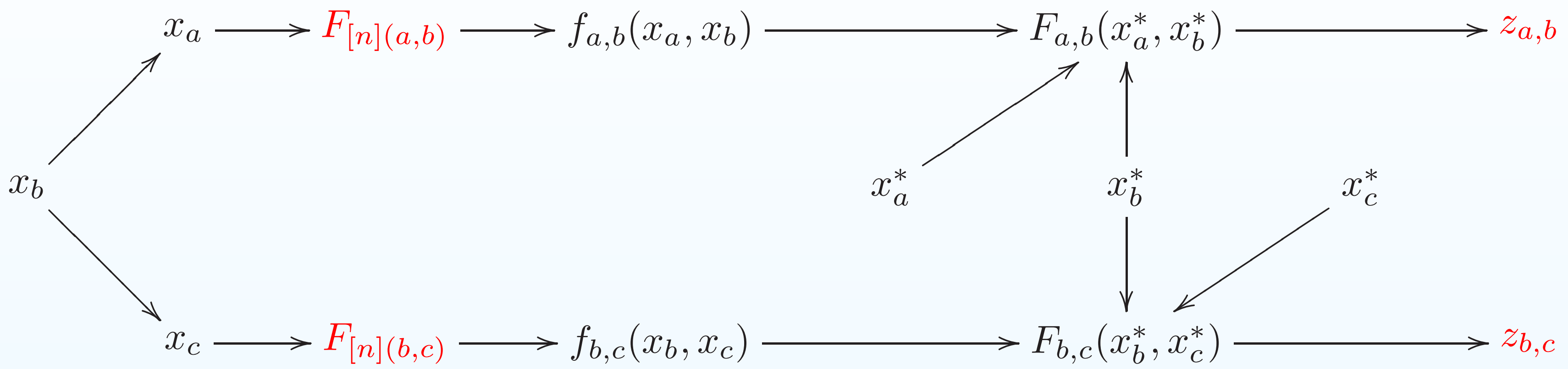
Evaluations, $F_{[n](a,b)}$ and $F_{[n](b,c)}$ are inputs to the corresponding emulators $f_{a,b}(x_a, x_b)$, $f_{b,c}(x_b, x_c)$

Emulation and best evaluations



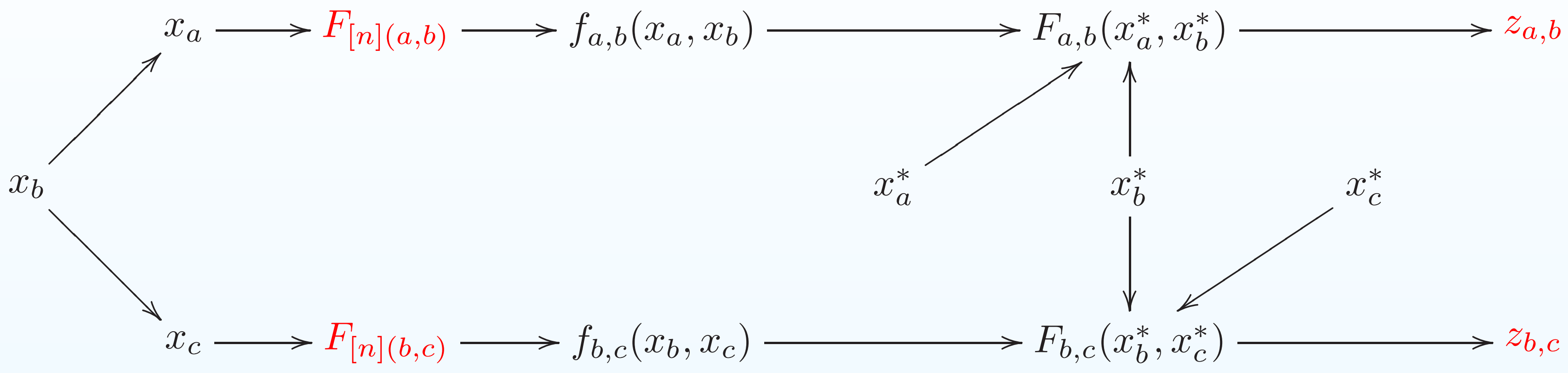
The emulators combine with the true values x^* to generate judgements for model runs at true inputs

Emulation and Calibration



The link to data observations z allows us to assess our implausibility measures over the input space, x, by local computation.

Emulation and Calibration



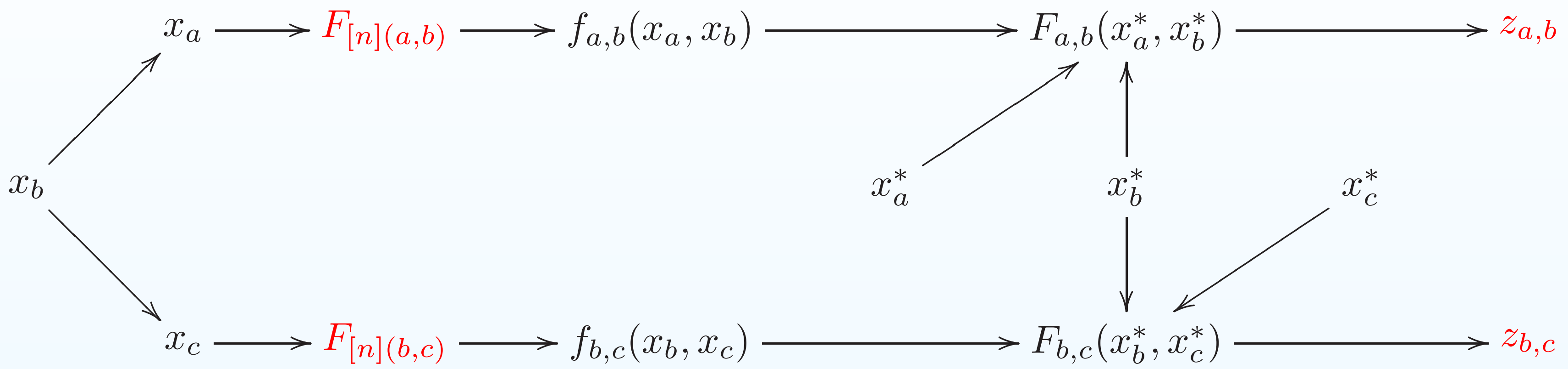
The link to data observations z allows us to assess our implausibility measures over the input space, x, by local computation.

In the above diagram, we collect the implausibility measure to x_b from

(i) the x_a, x_b pair, based on $z_{a,b}$

(ii) the x_b, x_c pair, based on $z_{b,c}$

Emulation and Calibration



The link to data observations z allows us to assess our implausibility measures over the input space, x, by local computation.

In the above diagram, we collect the implausibility measure to x_b from

(i) the x_a, x_b pair, based on $z_{a,b}$

(ii) the x_b, x_c pair, based on $z_{b,c}$

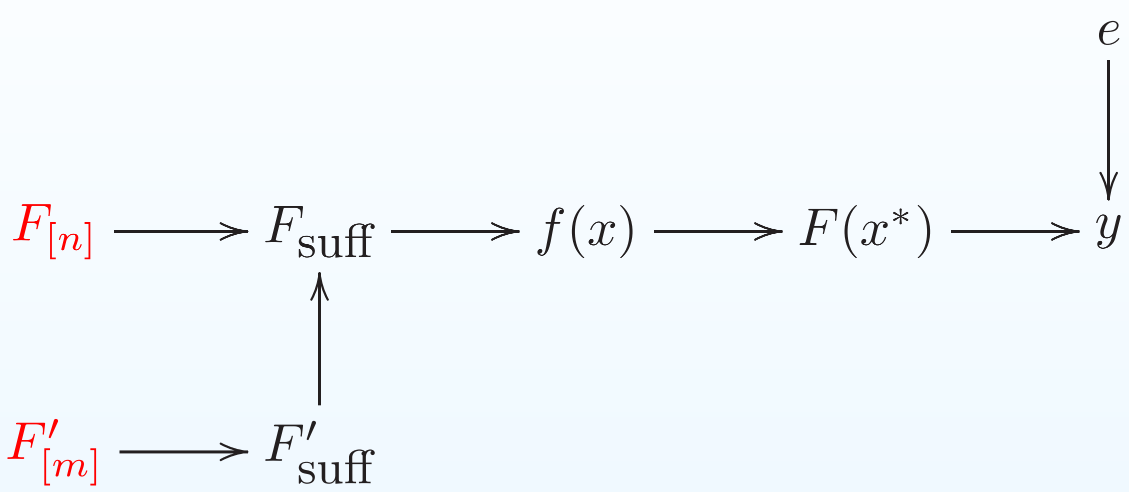
We then distribute the combined implausibility measure back to x_a and x_c.

Small samples

$$\begin{array}{ccccccccc} & & & & & & & & e \\ & & & & & & & & \downarrow \\ F_{[n]} & \longrightarrow & F_{\text{suff}} & \longrightarrow & f(x) & \longrightarrow & F(x^*) & \longrightarrow & y \end{array}$$

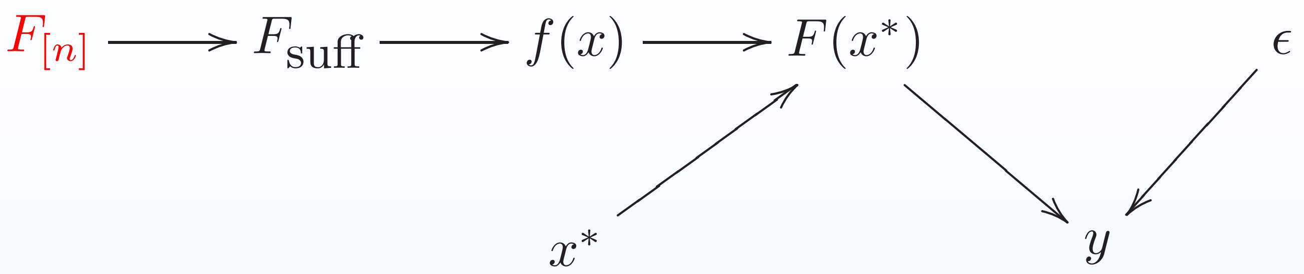
Often, we can only make a few evaluations of our computer simulator, so that our evaluation $F_{[n]}$ is based on small value of n.

Small samples and fast approximations



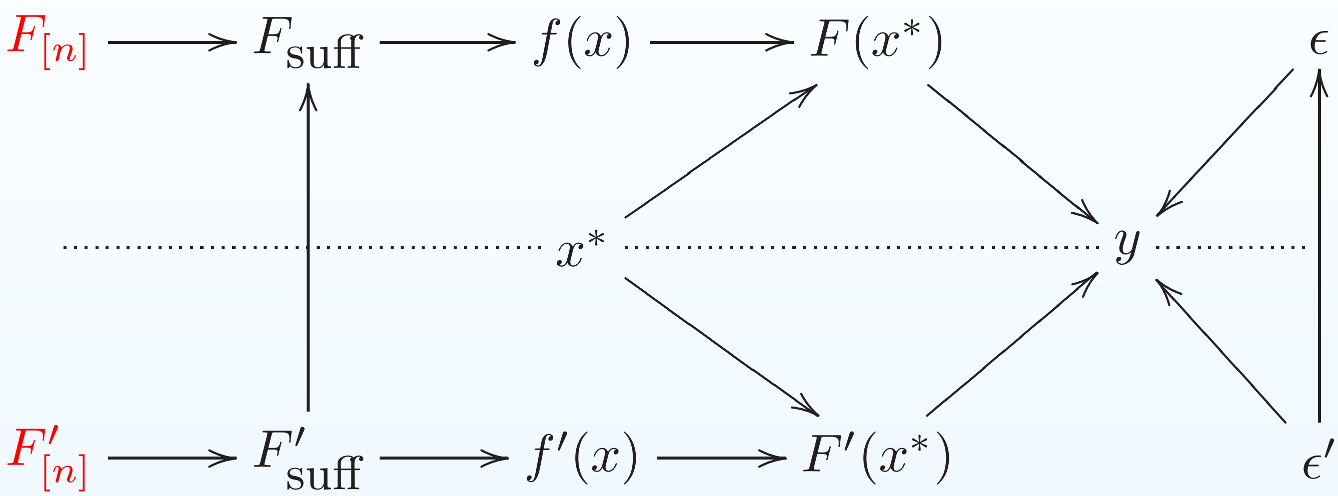
We may be able to make many evaluations, $F'_{[m]}$ of a simpler approximate version of the model as a basis for the inference.

A graphical puzzle



We link evaluations of our simulator F through our emulator to the system values.

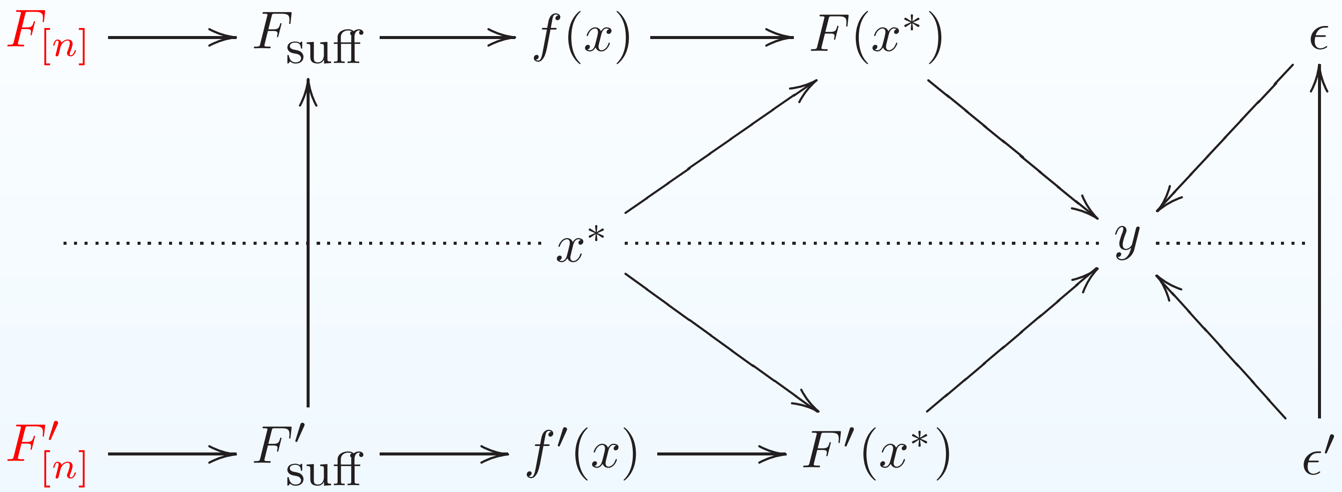
A graphical puzzle



Now add the fast approximation F' to the graph.

But suppose that, last year, the fast approximation was the full model, for which we had already drawn the corresponding version of this graph.

A graphical puzzle



Now add the fast approximation F' to the graph.

But suppose that, last year, the fast approximation was the full model, for which we had already drawn the corresponding version of this graph.

Comment: you can't get all of the conditional orthogonalities in the above diagram without imposing unreasonable constraints on the system.

Reification

What does a simulator F really tell us about a physical system y?

Reification

What does a simulator F really tell us about a physical system y?

How do we combine the information about y from a collection of simulators?

Reification

What does a simulator F really tell us about a physical system y?
How do we combine the information about y from a collection of simulators?

Consider both our inputs x and the simulator F as abstractions/simplifications of the way in which real system properties affect real system behaviour. Denote this more realistic simulator by F^*.

Reification

What does a simulator F really tell us about a physical system y?
How do we combine the information about y from a collection of simulators?

Consider both our inputs x and the simulator F as abstractions/simplifications of the way in which real system properties affect real system behaviour. Denote this more realistic simulator by F^*.
For F^*, the real, physical x^* would be the best input, in the sense that $(y - F^*(x^*))$ would be judged independent of (x^*, F^*).

Reification

What does a simulator F really tell us about a physical system y?
How do we combine the information about y from a collection of simulators?

Consider both our inputs x and the simulator F as abstractions/simplifications of the way in which real system properties affect real system behaviour. Denote this more realistic simulator by F^*.
For F^*, the real, physical x^* would be the best input, in the sense that $(y - F^*(x^*))$ would be judged independent of (x^*, F^*).

We call F^* the reified simulator (from reify: to treat an abstract concept as if it was real).

Reification

What does a simulator F really tell us about a physical system y?
How do we combine the information about y from a collection of simulators?

Consider both our inputs x and the simulator F as abstractions/simplifications of the way in which real system properties affect real system behaviour. Denote this more realistic simulator by F^*.
For F^*, the real, physical x^* would be the best input, in the sense that $(y - F^*(x^*))$ would be judged independent of (x^*, F^*).

We call F^* the reified simulator (from reify: to treat an abstract concept as if it was real).

Reifying principle
[1] Simulator F is informative for y, because F is informative for F^* and $F^*(x^*)$ is informative for y.

Reification

What does a simulator F really tell us about a physical system y?
How do we combine the information about y from a collection of simulators?

Consider both our inputs x and the simulator F as abstractions/simplifications of the way in which real system properties affect real system behaviour. Denote this more realistic simulator by F^*.
For F^*, the real, physical x^* would be the best input, in the sense that $(y - F^*(x^*))$ would be judged independent of (x^*, F^*).

We call F^* the reified simulator (from reify: to treat an abstract concept as if it was real).

Reifying principle
[1] Simulator F is informative for y, because F is informative for F^* and $F^*(x^*)$ is informative for y.
[2] A collection of simulators $F_1, F_2, \ldots$ is jointly informative for y, as the simulators are jointly informative for F^*.

Linking F and F^* using emulators

Suppose that our emulator for F is

$$f(x) = Bg(x) + u(x)$$

Linking F and F^* using emulators

Suppose that our emulator for F is

$$f(x) = Bg(x) + u(x)$$

Our simplest emulator for F^* might be

$$f^*(x, w) = B^*g(x) + u^*(x) + u^*(x, w)$$

where we might model our judgements as $B^* = CB + \Gamma$, correlate $u(x)$ and $u^*(x)$, while $u^*(x, w)$, with additional parameters, w, is uncorrelated with remainder.

Linking F and F^* using emulators

Suppose that our emulator for F is

$$f(x) = Bg(x) + u(x)$$

Our simplest emulator for F^* might be

$$f^*(x, w) = B^*g(x) + u^*(x) + u^*(x, w)$$

where we might model our judgements as $B^* = CB + \Gamma$, correlate $u(x)$ and $u^*(x)$, while $u^*(x, w)$, with additional parameters, w, is uncorrelated with remainder.

Structured reification: systematic probabilistic modelling for all those aspects of model deficiency whose effects we are prepared to consider explicitly.

Reified inference structure

$$F_{[n]} \longrightarrow F_{\text{suff}}$$

$F_{[n]}$: n evaluations of F at inputs $x_1, x_2, \ldots$
F_{suff}: the global information from $F_{[n]}$.

Reified inference structure

$$F_{[n]} \longrightarrow F_{\text{suff}} \longrightarrow F^*_{\text{suff}}$$

F^*_{suff}: corresponding global information for reified emulator $f^*(x)$

Reified inference structure

$$\begin{array}{cccccc} & & & & x^* & \epsilon \\ & & & & \downarrow & \downarrow \\ F_{[n]} \longrightarrow & F_{\text{suff}} \longrightarrow & F^*_{\text{suff}} \longrightarrow & f^*(x) \longrightarrow & F^*(x^*) \longrightarrow & y \end{array}$$

True system properties x^* with emulator $f^*(x)$ influence beliefs for $F(x^*)$, which is informative for system values y, with discrepancy ϵ.

Reified inference structure

$$\begin{array}{ccccccccccc} & & & & & & & & x^* & & \epsilon \\ & & & & & & & & \downarrow & & \downarrow \\ F_{[n]} & \longrightarrow & F_{\text{suff}} & \longrightarrow & F^*_{\text{suff}} & \longrightarrow & f^*(x) & \longrightarrow & F^*(x^*) & \longrightarrow & y \end{array}$$

True system properties x^* with emulator $f^*(x)$ influence beliefs for $F(x^*)$, which is informative for system values y, with discrepancy ϵ.

Comment: All our calibration and forecasting methodology is unchanged - all that has changed is our description of the joint covariance structure.

A Reified influence diagram

$$\left[F^{1}_{h:[n]}(x), \ldots, F^{m}_{h:[n]}(x)\right]$$

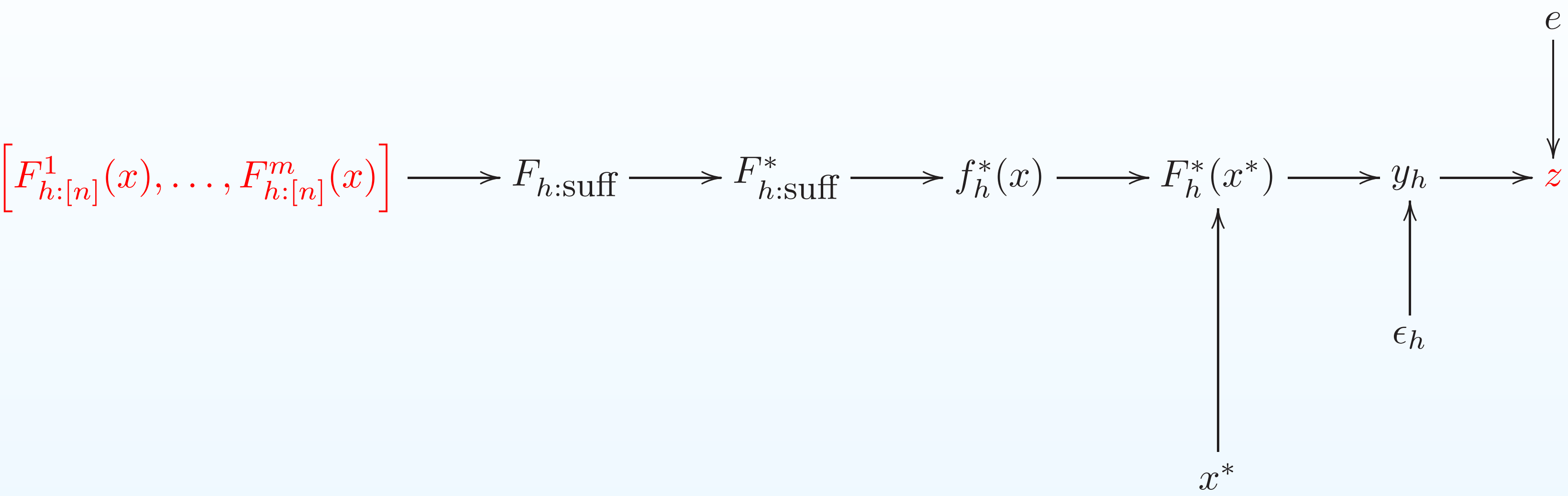
Evaluations of the simulator at each of m initial conditions
for historical components of simulator

A Reified influence diagram

$$\left[F^1_{h:[n]}(x), \ldots, F^m_{h:[n]}(x)\right] \longrightarrow F_{h:\mathrm{suff}} \longrightarrow F^*_{h:\mathrm{suff}} \longrightarrow f^*_h(x)$$

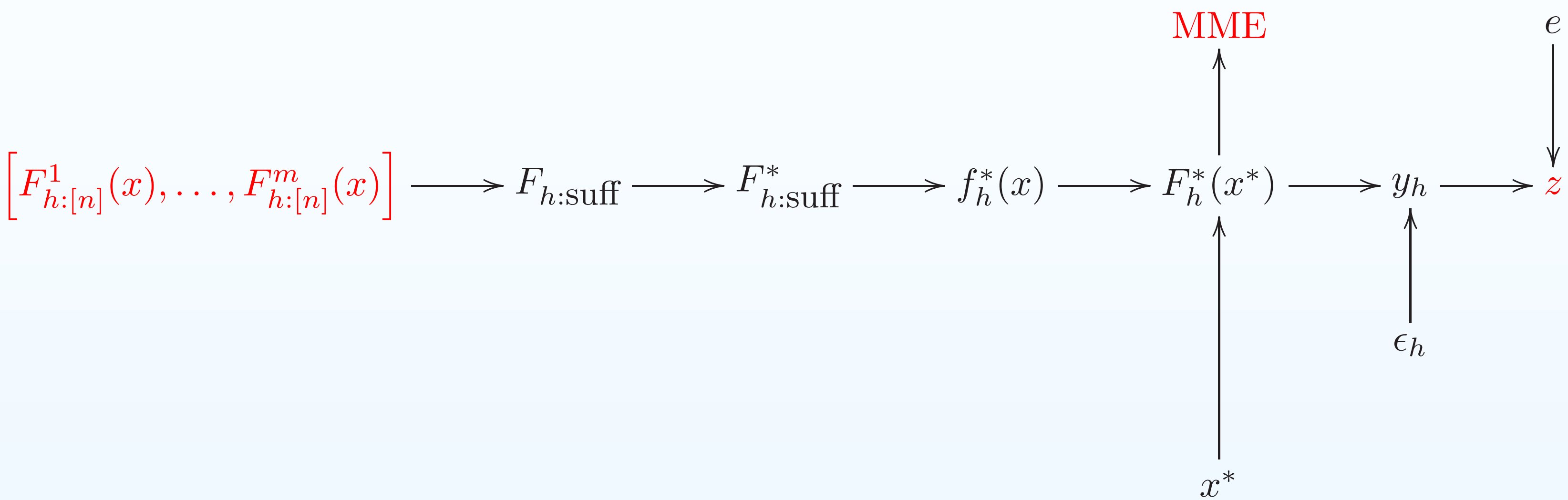
Global information $F_{\mathrm{h:suff}}$ (from second order exchangeability modelling).
passes to Reified global form and to reified emulator.

A Reified influence diagram



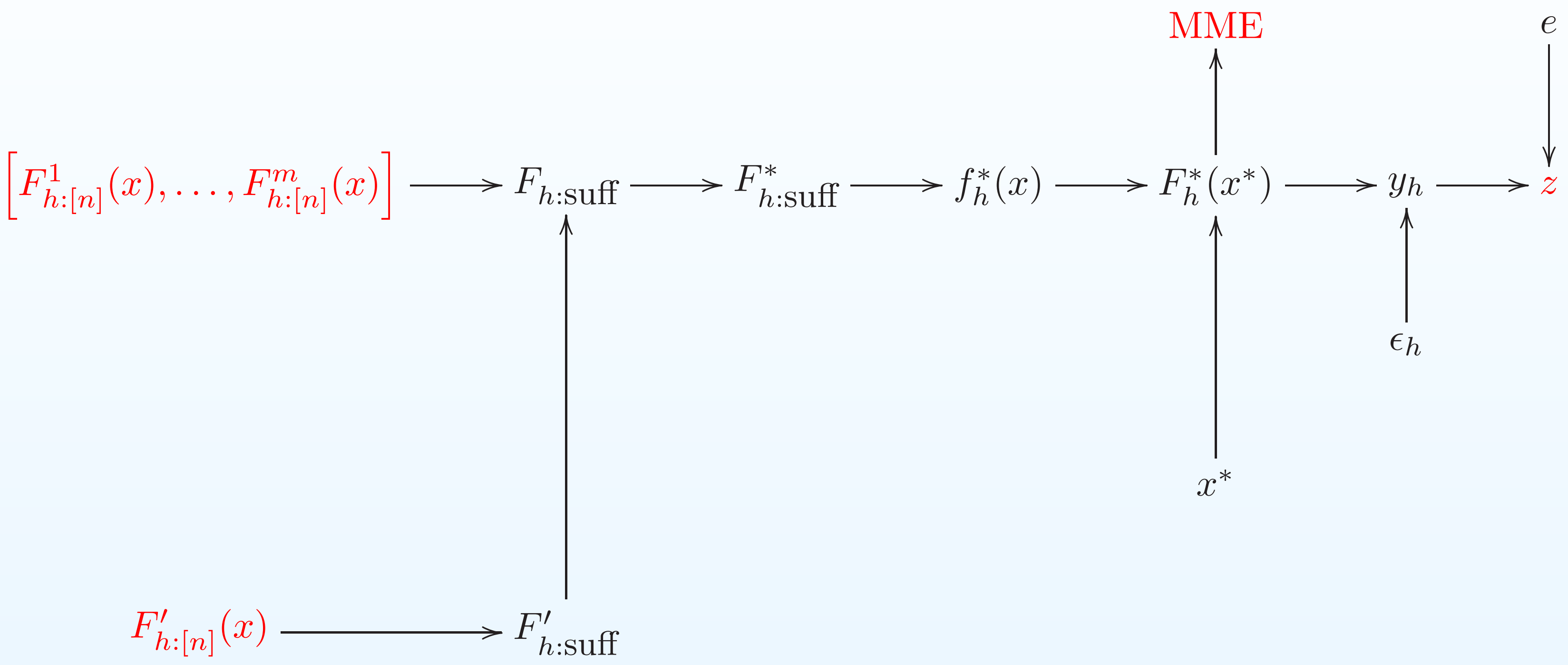
Link with x^* to reified function, at true initial condition, linked to data z

A Reified influence diagram



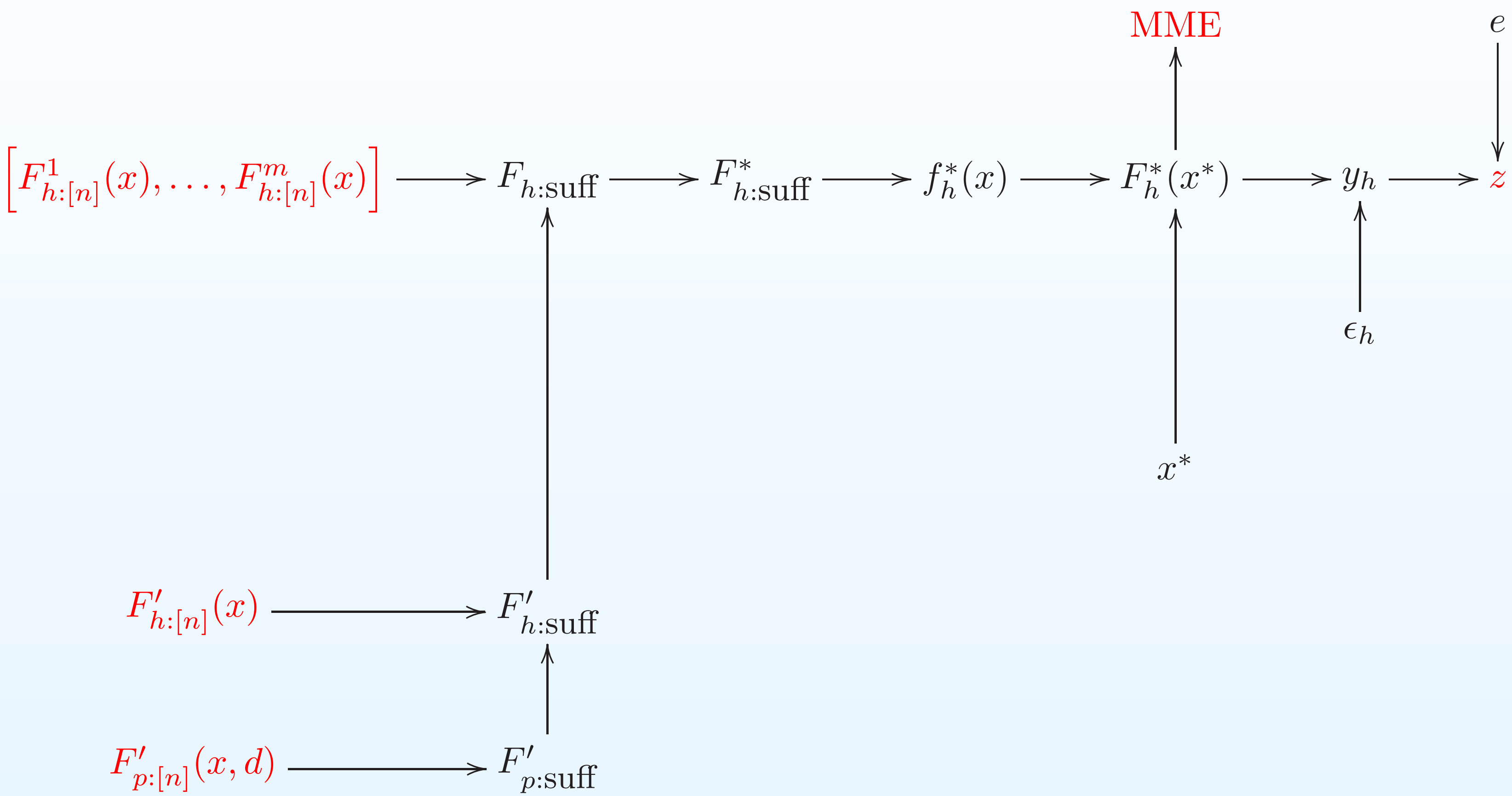
Add observation of a related multi-model ensemble (MME) consisting of tuned runs from related models (more exchangeability modelling).

A Reified influence diagram



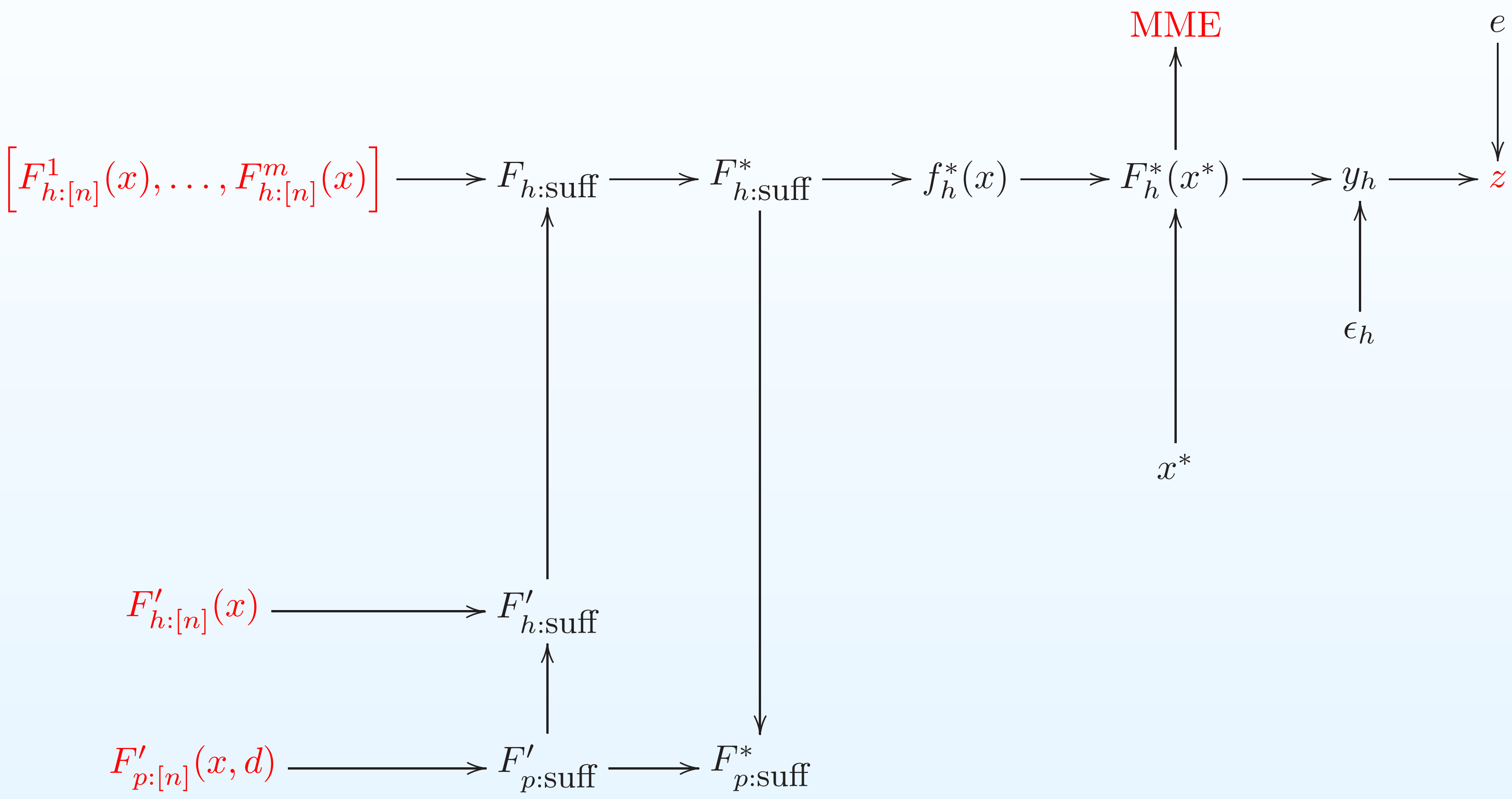
Add a set of evaluations from a fast approximation

A Reified influence diagram



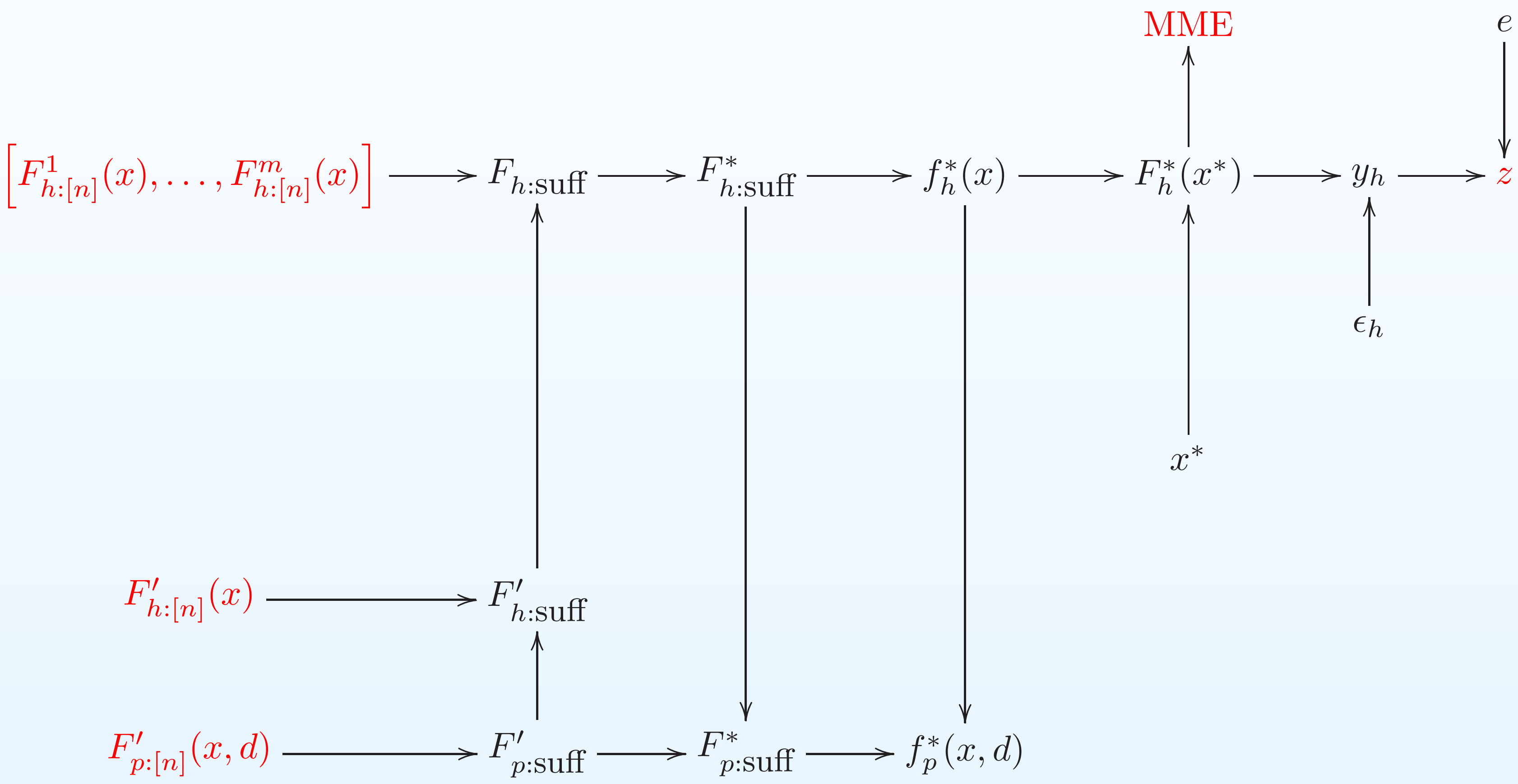
Add evaluations of fast simulator for outcomes to be predicted, with decision choices d

A Reified influence diagram



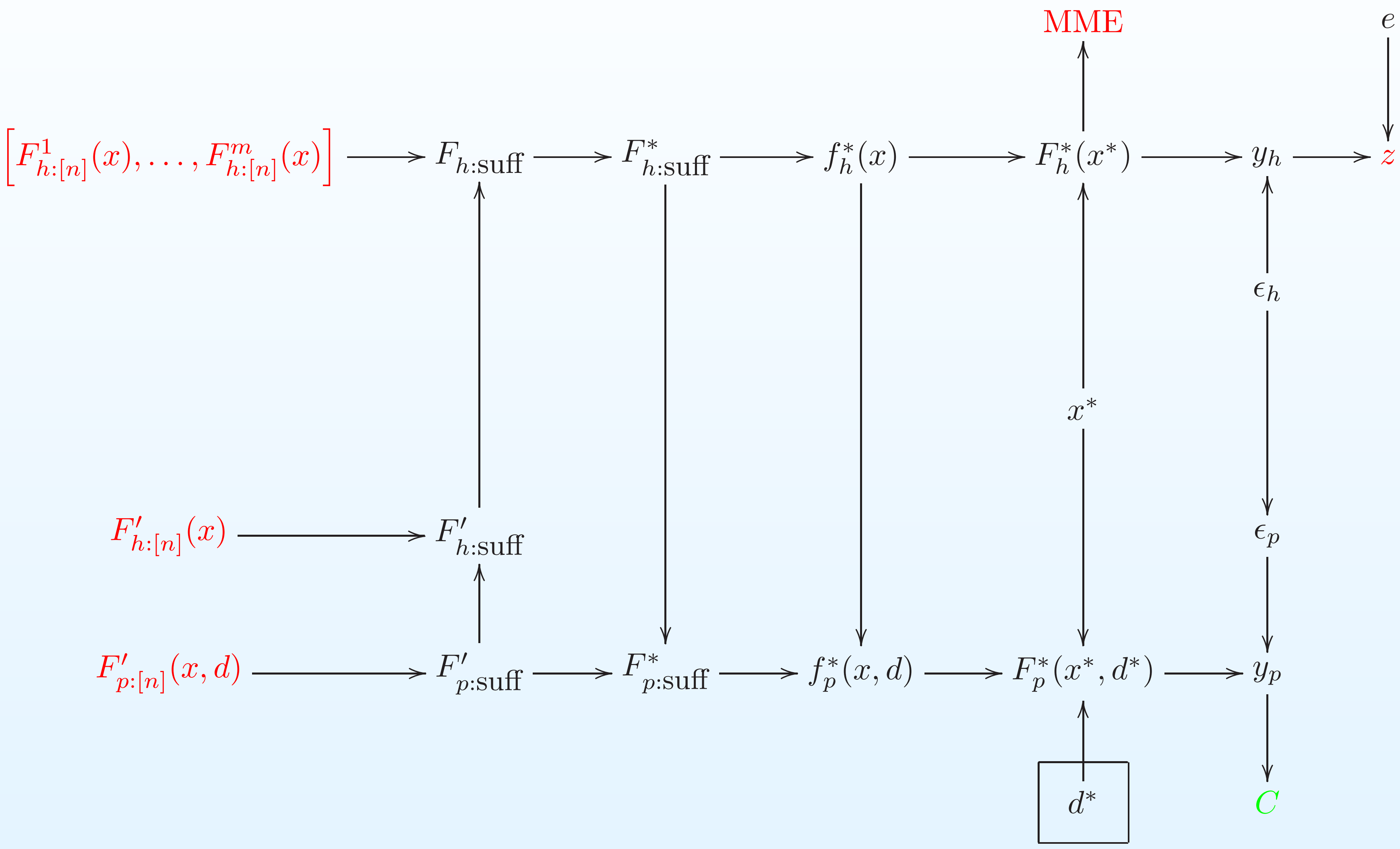
Link to reified global terms for quantities to be predicted

A Reified influence diagram



And to reified global emulator, based on inputs and decisions

A Reified influence diagram



And link, through true future values y_p, to the overall utility cost C of making decision choice d^*.

Best current judgements for complex systems

To assess best current judgements about complex systems, it is enormously helpful to have an overall framework to unify all the uncertainties arising from

Uncertain model parameters, outputs and discrepancies

Uncertain observations/initial conditions/forcing functions

Uncertain relationships between different modelling approaches

Uncertain effects of our attempts to influence the system

Best current judgements for complex systems

To assess best current judgements about complex systems, it is enormously helpful to have an overall framework to unify all the uncertainties arising from

Uncertain model parameters, outputs and discrepancies

Uncertain observations/initial conditions/forcing functions

Uncertain relationships between different modelling approaches

Uncertain effects of our attempts to influence the system

Bayes linear influence diagrams provide a conceptual/graphical framework for unifying our qualitative and quantitative knowledge about all such uncertainties within a structure which is both logical and tractable, so that we can focus on science rather than technical/computational issues.

Best current judgements for complex systems

To assess best current judgements about complex systems, it is enormously helpful to have an overall framework to unify all the uncertainties arising from

Uncertain model parameters, outputs and discrepancies

Uncertain observations/initial conditions/forcing functions

Uncertain relationships between different modelling approaches

Uncertain effects of our attempts to influence the system

Bayes linear influence diagrams provide a conceptual/graphical framework for unifying our qualitative and quantitative knowledge about all such uncertainties within a structure which is both logical and tractable, so that we can focus on science rather than technical/computational issues.

Such analysis poses serious challenges, but they are no harder than all of the other modelling, computational and observational challenges involved with studying large scale physical systems.

References

P.S. Craig, M. Goldstein, J.C.Rougier, A.H. Seheult, (2001) Bayesian Forecasting Using Large Computer Models, JASA, 96, 717-729

M. Goldstein and J.C.Rougier (2009). Reified Bayesian modelling and inference for physical systems (with discussion), JSPI, 139, 1221-1239

M. Goldstein and D.A.Wooff (2007) Bayes linear statistics: theory and methods (esp. Chapter 10) Wiley

References

P.S. Craig, M. Goldstein, J.C.Rougier, A.H. Seheult, (2001) Bayesian Forecasting Using Large Computer Models, JASA, 96, 717-729

M. Goldstein and J.C.Rougier (2009). Reified Bayesian modelling and inference for physical systems (with discussion), JSPI, 139, 1221-1239

M. Goldstein and D.A.Wooff (2007) Bayes linear statistics: theory and methods (esp. Chapter 10) Wiley

And check out the website for the
Managing Uncertainty in Complex Models (MUCM) project
[A consortium of Aston, Durham, LSE, Sheffield and Southampton all hard at work on developing technology for computer model uncertainty problems.]