Figuring out why your model is wrong

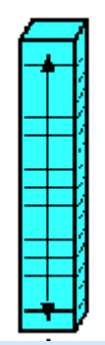
Vincent E. Larson

Dept. of Math Sciences
University of Wisconsin --- Milwaukee

I acknowledge my collaborators Jim Hansen, Brian Griffin, David Schanen, and especially Chris Golaz, who produced many of the figures.

Outline

- Overview: Diagnosing model errors using calibration.
- Description of our single-column model (parameterization) for boundary layer clouds and turbulence.



- Our calibration methodology
- An example: Diagnosing errors in our single-column model.
- Can any of this aid superparameterization?

Diagnosing the source of model error is a widespread problem

Every physical model, from every branch of science, has errors.

Before we can improve a model, we need to figure out what is wrong with it.

We divide model error into two classes: Structural model error, and an error in a parameter value

Structural model error is an error in the functional form of a term. For instance, perhaps a frictional drag term should have, hypothetically, quadratic damping

$$-(Constant) \times u^2/\tau,$$

not linear damping

$$-(Constant) \times u/\tau.$$

Parametric error is an error in the specification of an adjustable parameter. For instance, perhaps linear damping is correct, but *Constant* should be set to 1.5, not 1.2.

Diagnosing structural model error is tough!

Many models consist of tightly coupled, nonlinear equations.

Errors propagate rapidly from term to term, infecting all fields.

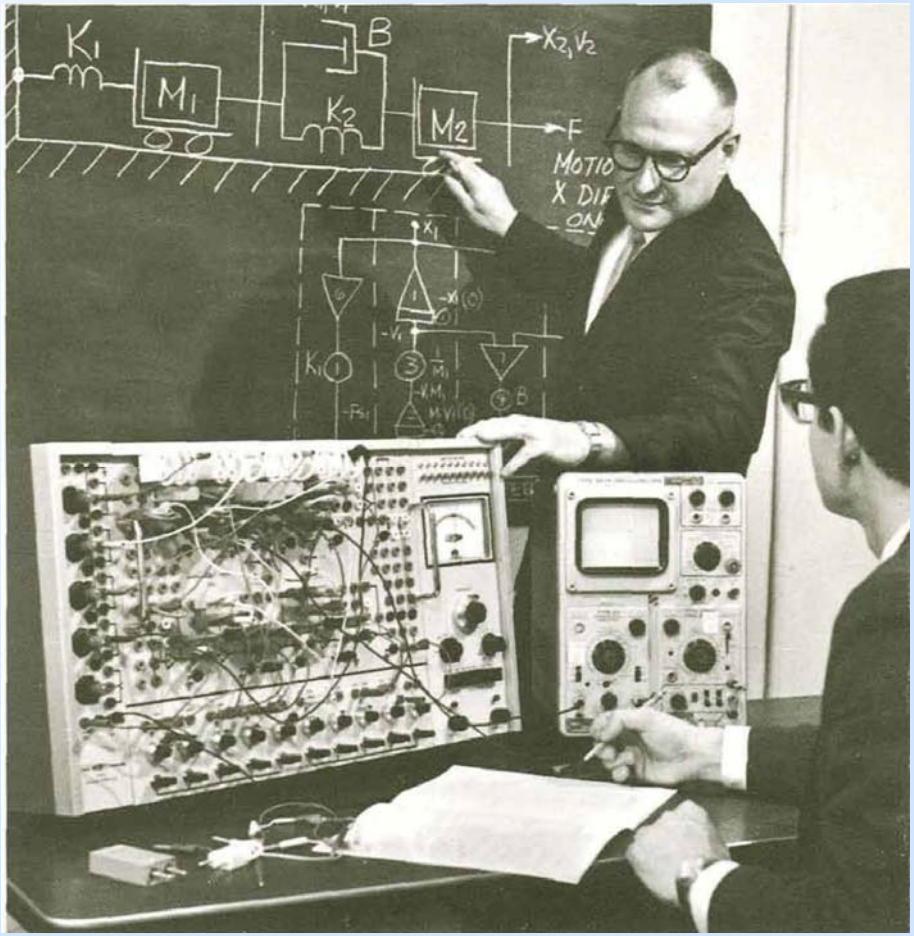
To fix the errors, we need to isolate the ultimate sources of them, not merely the symptoms.

To diagnose model errors, the community is using the mathematical equivalent of patch cables!

It's analogous software development.

To write rudimentary codes, one may use patch cables. For more complex codes, one must develop supporting tools, such as programming languages, subroutines, and debuggers.

Similarly, for diagnosing model errors, we need a mathematical meta-tool to help sort through the complexity.



CG

One meta-tool that may help diagnose model error is calibration

Calibration = optimization of adjustable parameters = "parameter estimation"

Calibration is usually used to improve model fit to data. This is a good thing, of course.

However, calibration has a sordid reputation because people suspect that calibration has merely hid structural errors. "Calibration covers a multitude of sins."

It need not. Quite oppositely, we aim to use calibration to maximally uncover structural errors.

The basic idea: Calibrate separately to *two physically different* datasets, and compare the two sets of optimized parameter values.

The two datasets could be, for instance, cumulus (Cu) and stratocumulus (Sc) regimes, which are fundamentally different. We desire a generalized model that works for both.

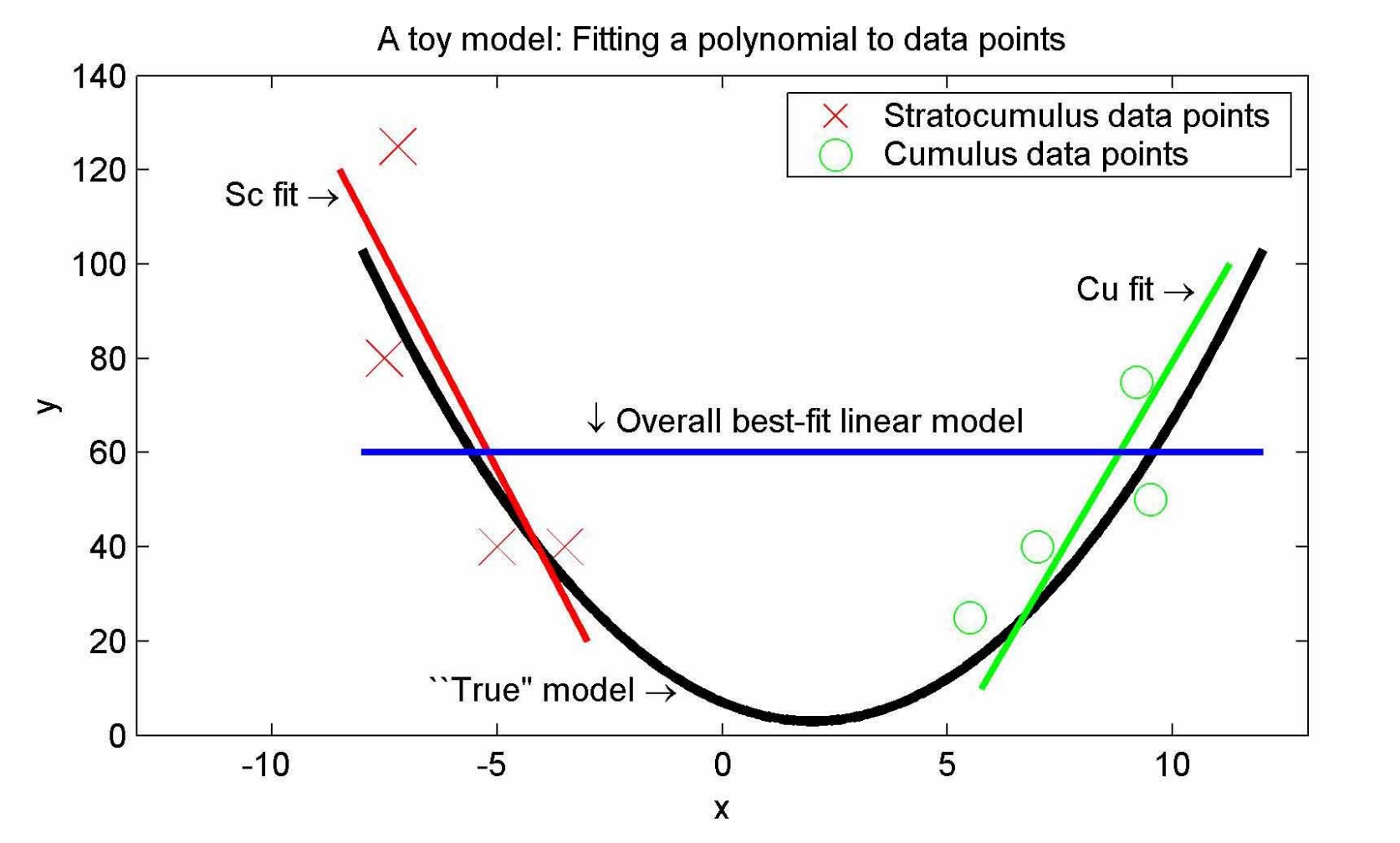
If the 2 parameter sets differ, it suggests there is a structural error and hints at its source.

If the fit in each calibration is improved, and *the two sets of parameter values are similar*, it suggests that there was parametric error.

Therefore, calibration distinguishes parametric from structural error.

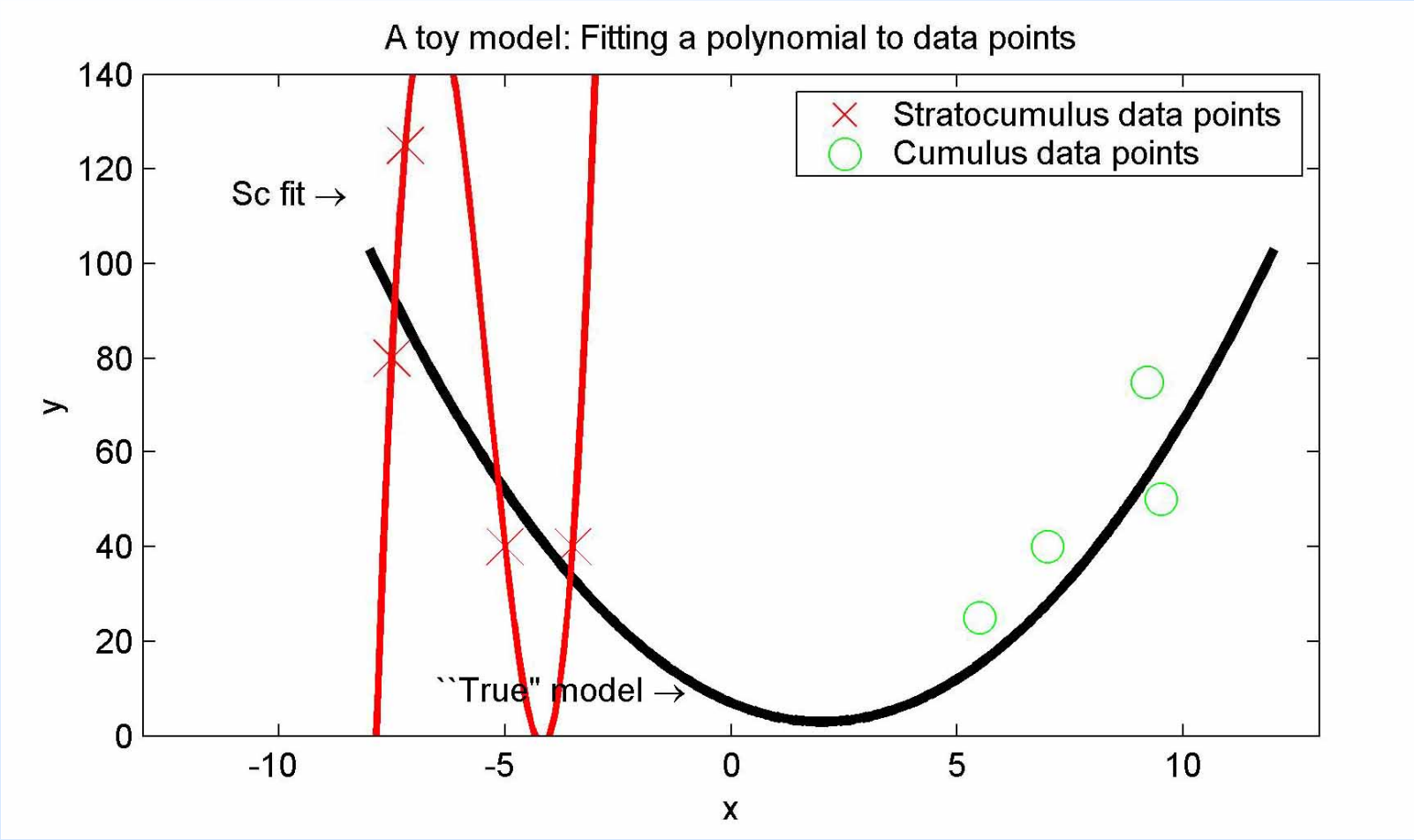
One could dream up many variants of this idea. For instance, the datasets or calibrated parameter sets may be varied.

One type of model error: underfitting



Underfitting can arise from model structural error, i.e. “not enough” parameters.

A hazard of calibration: overfitting



Overfitting arises from having too many parameters and too few data. If a model is overfit, it is unlikely to work for new (i.e. out-of-sample) cases.

The evil is not calibration per se, but overfitting

Overfitting may be detected by cross-validation.

In cross-validation, a model is fitted to one dataset, and tested against an independent dataset.

In the past, obtaining and analyzing several independent datasets was difficult; today it is quite feasible.

Calibration does not deserve a bad reputation

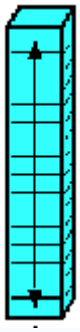
Note an analogy with neural nets. The equations in a neural net contain no physics, only adjustable parameters.

Neural net	**Physical model w/ tunable parameters**
"Weights"	Adjustable parameters
"Learning/Training"	Calibrating
"Memorizing"	Overfitting

And yet with enough training data, neural nets work well for some problems. Surely, adding physics can only improve the performance.

Outline

- Overview: Diagnosing model errors using calibration.
- Description of our single-column model for boundary layer clouds and turbulence.
- Our calibration methodology
- An example: Diagnosing errors in our single-column model.
- Can any of this aid superparameterization?



Our single-column model (SCM) is a model of boundary layer clouds and turbulence

Why do we focus on clouds?

"Clouds provide the largest source of uncertainty in current model predictions of climate sensitivity."
--- Soden and Held (2006)

Why ***boundary layer*** clouds?

"Marine boundary layer clouds are at the heart of tropical cloud feedback uncertainties in climate models." --- Bony and Dufresne (2005)

Low clouds "are responsible for 59% of the contribution of inter-model differences in the net cloud feedback". --- Webb et al. (2006)

The "source of disagreement between [GFDL AM and NCAR CAM] still seems to be the low cloud effects".
--- Medeiros et al. (2007)

Our SCM predicts 1D profiles that depend on altitude (*z*) and time (*t*)

Our single-column model predicts cloud fraction *cf*, liquid water mixing ratio q_c, and many other variables as a function of *z* and *t*:

$$
\begin{aligned}
cf &= cf(z,t) \\
q_c &= q_c(z,t) \\
&\ldots
\end{aligned}
$$

Our SCM is deterministic, not stochastic.

Our SCM may be interpreted as a set of higher-order moment equations with a novel closure

The complete set of prognosed moments is:

$$\text{Means}: \quad \frac{\partial \overline{u}}{\partial t} = \ldots \quad \frac{\partial \overline{v}}{\partial t} = \ldots \quad \frac{\partial \overline{q_t}}{\partial t} = \ldots \quad \frac{\partial \overline{\theta_l}}{\partial t} = \ldots$$

$$\text{2nd} - \text{order}: \quad \frac{\partial \overline{w' q_t'}}{\partial t} = \ldots \quad \frac{\partial \overline{w' \theta_l'}}{\partial t} = \ldots \quad \frac{\partial \overline{w'^2}}{\partial t} = \ldots$$

$$\frac{\partial \overline{q_t'^2}}{\partial t} = \ldots \quad \frac{\partial \overline{\theta_l'^2}}{\partial t} = \ldots \quad \frac{\partial \overline{q_t' \theta_l'}}{\partial t} = \ldots$$

$$\text{3rd} - \text{order}: \quad \frac{\partial \overline{w'^3}}{\partial t} = \ldots$$

Golaz et al. (2002)

w = vertical velocity $\quad q_t$ = total water specific humidity

θ_l = liquid water potential temperature

Various adjustable parameters (C_2, C_7, C_{11}, etc.) appear on the right-hand sides of these equations.

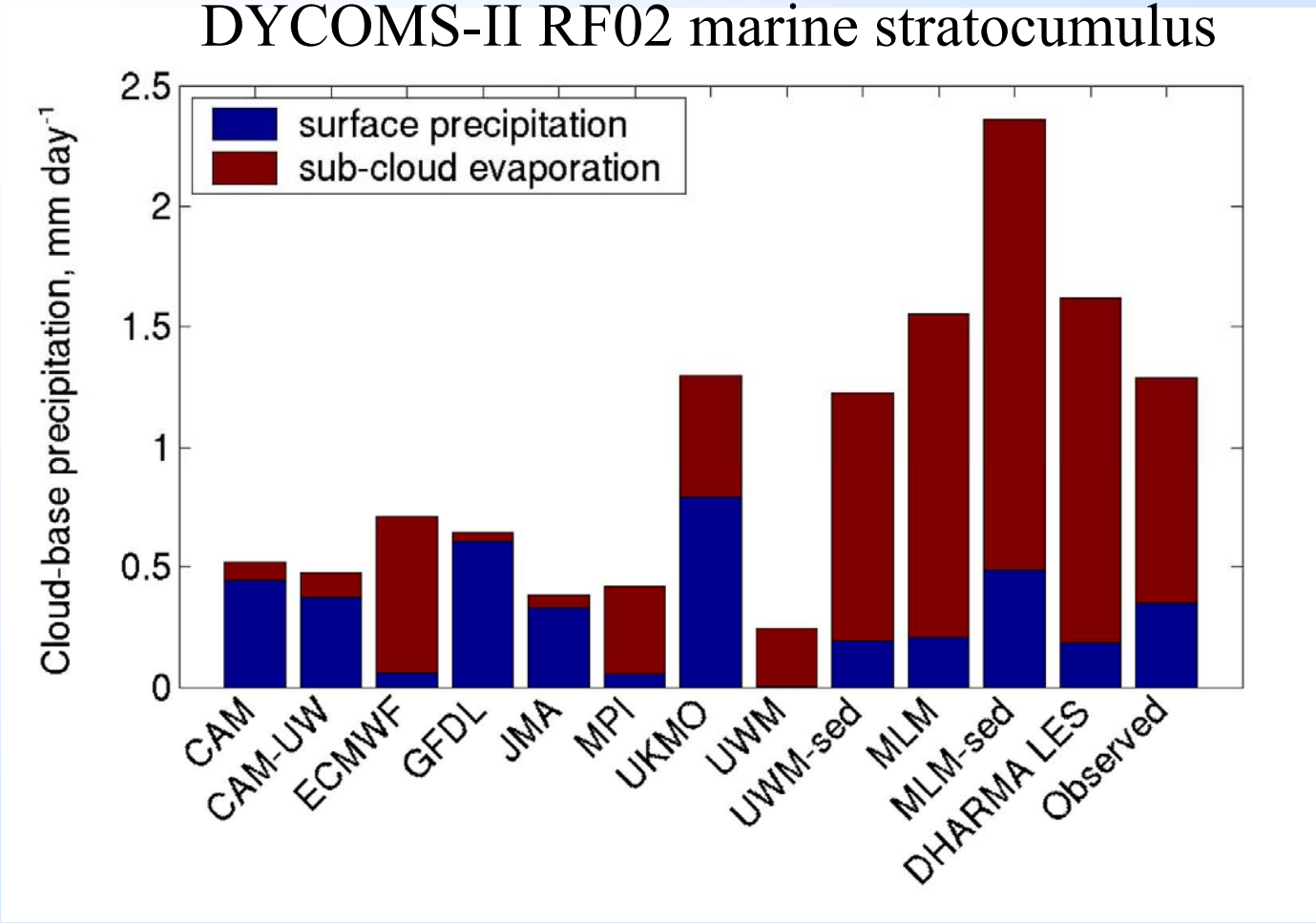
Our SCM has been used to simulate 11 different cases

These are:

1.	BOMEX	Trade-wind cumulus
2.	DYCOMS-II RF01	Marine stratocumulus
3.	FIRE	Marine stratocumulus
4.	DYCOMS-II RF02	Marine stratocumulus with drizzle
5.	ARM	Continental cumulus
6.	ATEX	Cumulus rising into stratocumulus
7.	MPACE	Arctic stratus
8.	RICO	Drizzling cumulus
9.	Wangara	Clear convective boundary layer
10.	Nov 11 CLEX-5	Altostratocumulus (mid level)
11.	GABLS II	Stable boundary layer

This talk will discuss the BOMEX Cu and RF01 Sc cases in detail, the black cases in passing, and the grey cases not at all.

Our SCM has been extended to include drizzle

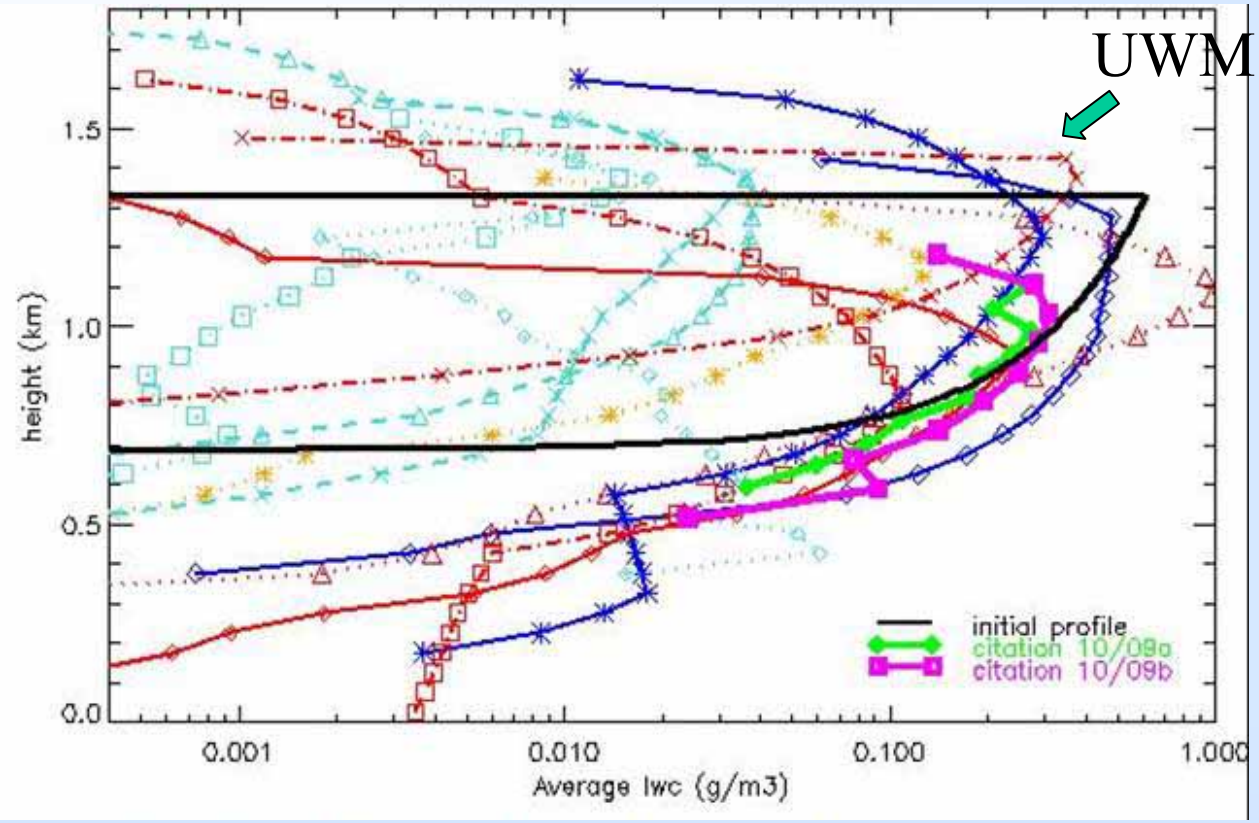


We interfaced the Khairoutdinov-Kogan drizzle parameterization by integrating it analytically over the subgrid variability.

Wyant et al. (2007)

We have also modeled an Arctic stratus cloud observed during MPACE

Liquid Water Content of SCMs



Intercomparison led by Steve Klein

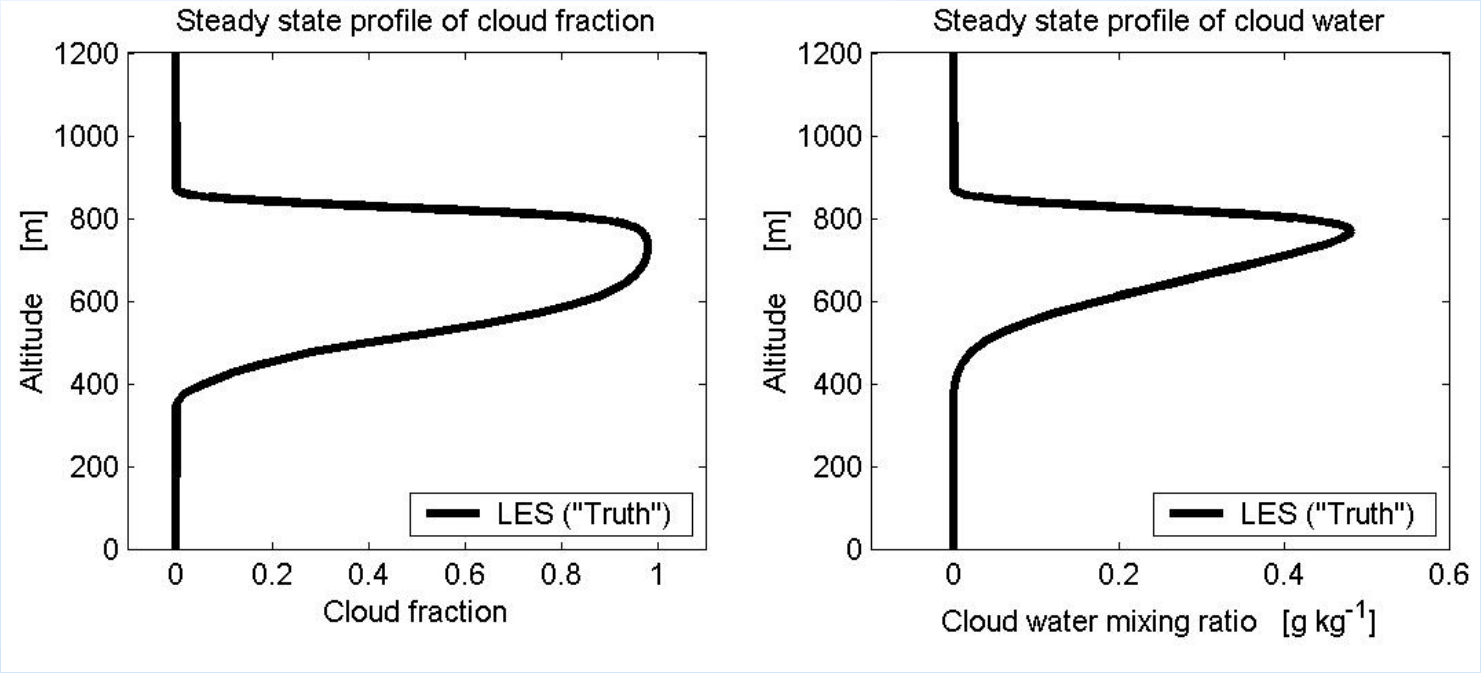
Outline

- Overview: Diagnosing model errors using calibration.
- Description of our single-column model for boundary layer clouds and turbulence.
- Our calibration methodology
- An example: Diagnosing errors in our single-column model.
- Can any of this aid superparameterization?

The “data” or “truth” that we strive to match is large-eddy simulation (LES) output

We perform large-eddy simulations of cumulus (Cu) and stratocumulus (Sc) cases. If we restrict our goal to emulating LES, then our data errors may be regarded as small.

The cases that we test are in a statistically steady state. Therefore we match time-averaged profiles of, e.g., cloud fraction and liquid water.

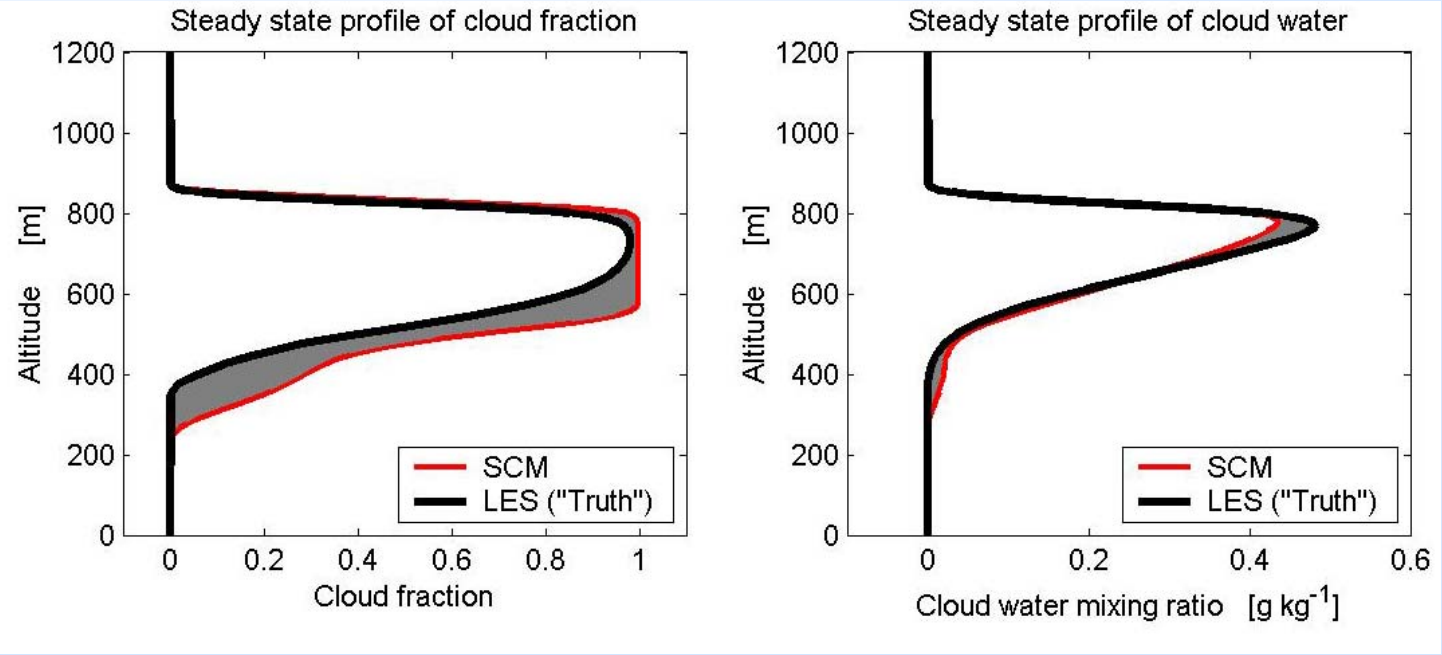


The agreement between "truth" and our SCM simulations is measured by a cost function, *J*

We strive to minimize *J*, where *J* is schematically given by

$$J \sim (\text{Cloud frac}_{SCM} - \text{Cloud frac}_{LES})^2 + (\text{Liquid}_{SCM} - \text{Liquid}_{LES})^2$$

J is essentially the square of the grey-shaded regions below:

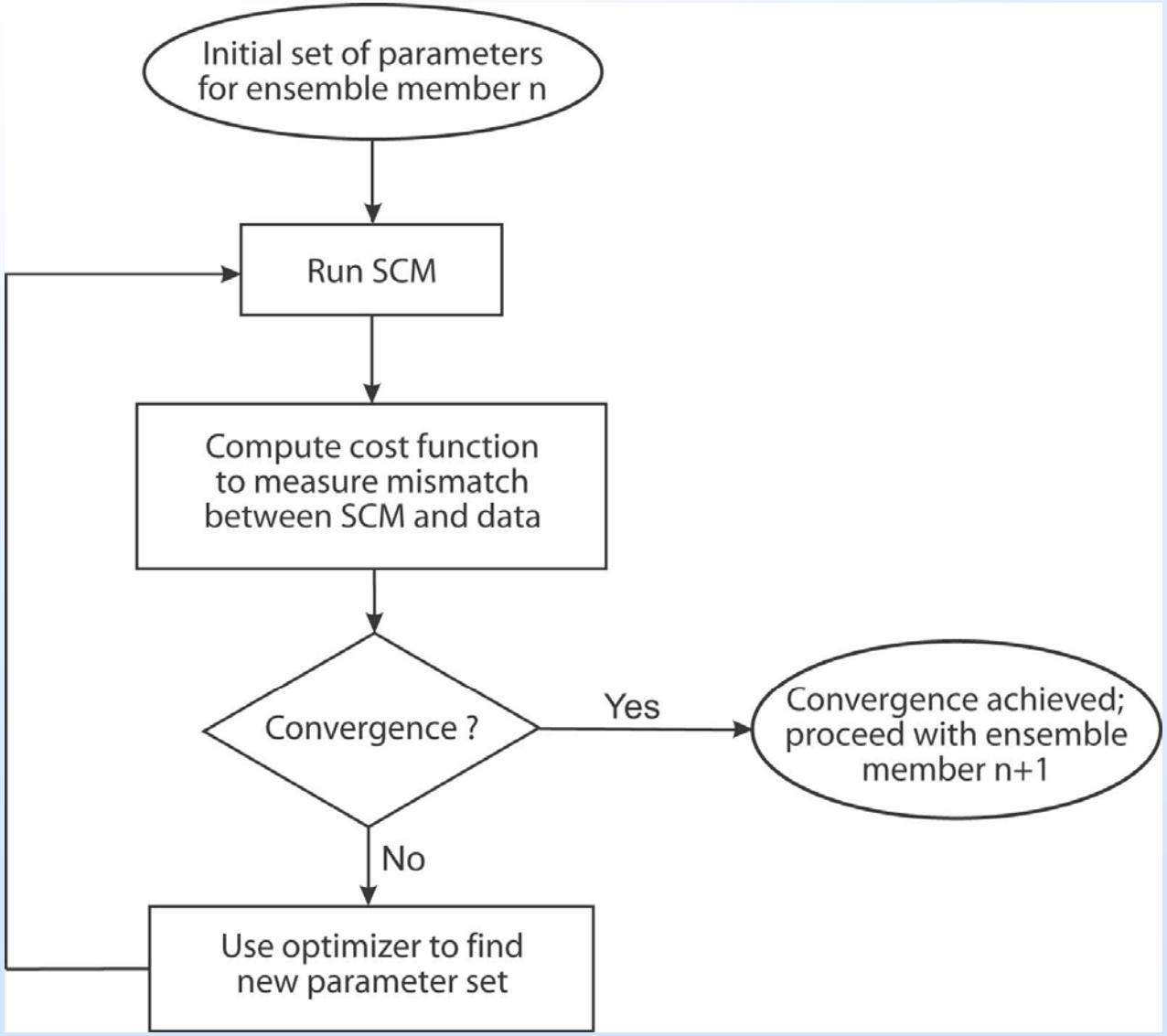


Our calibration technique is an ensemble technique

We do not strive to find a single set of optimal parameter values, but many sets of near-optimal parameter values.

One advantage: the ensemble spread provides uncertainty estimates of parameter values.

The steps in our calibration algorithm



… then repeat for the next ensemble member with different random initial parameter values, and so on . . .

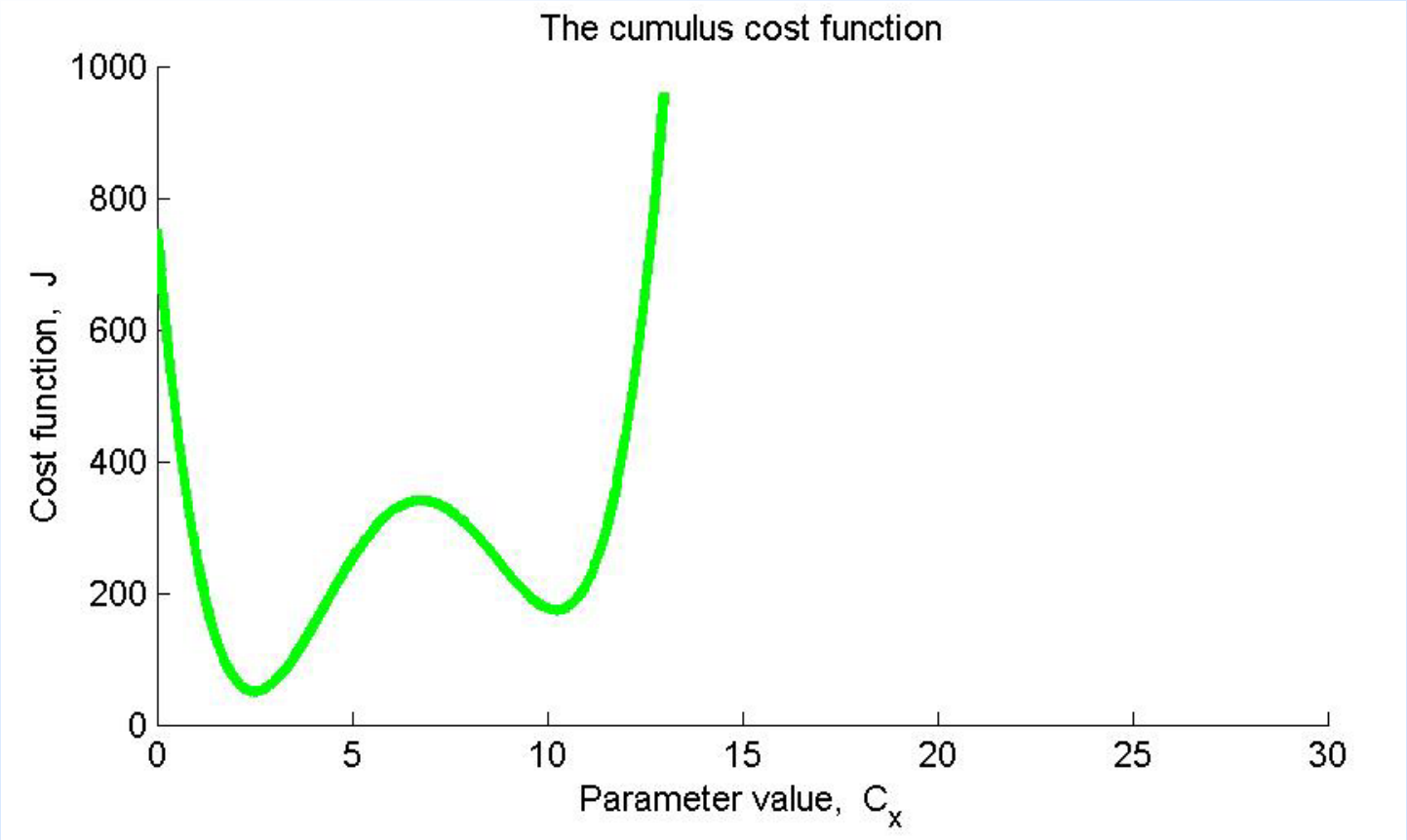
This produces an *ensemble* of sets of parameter values

1st Ensemble member :	[C_1	C_2	C_5	C_6	C_7	C_8	C_{11}]
				...			
*n*th Ensemble member :	[C_1	C_2	C_5	C_6	C_7	C_8	C_{11}]
				...			
Last Ensemble member :	[C_1	C_2	C_5	C_6	C_7	C_8	C_{11}]

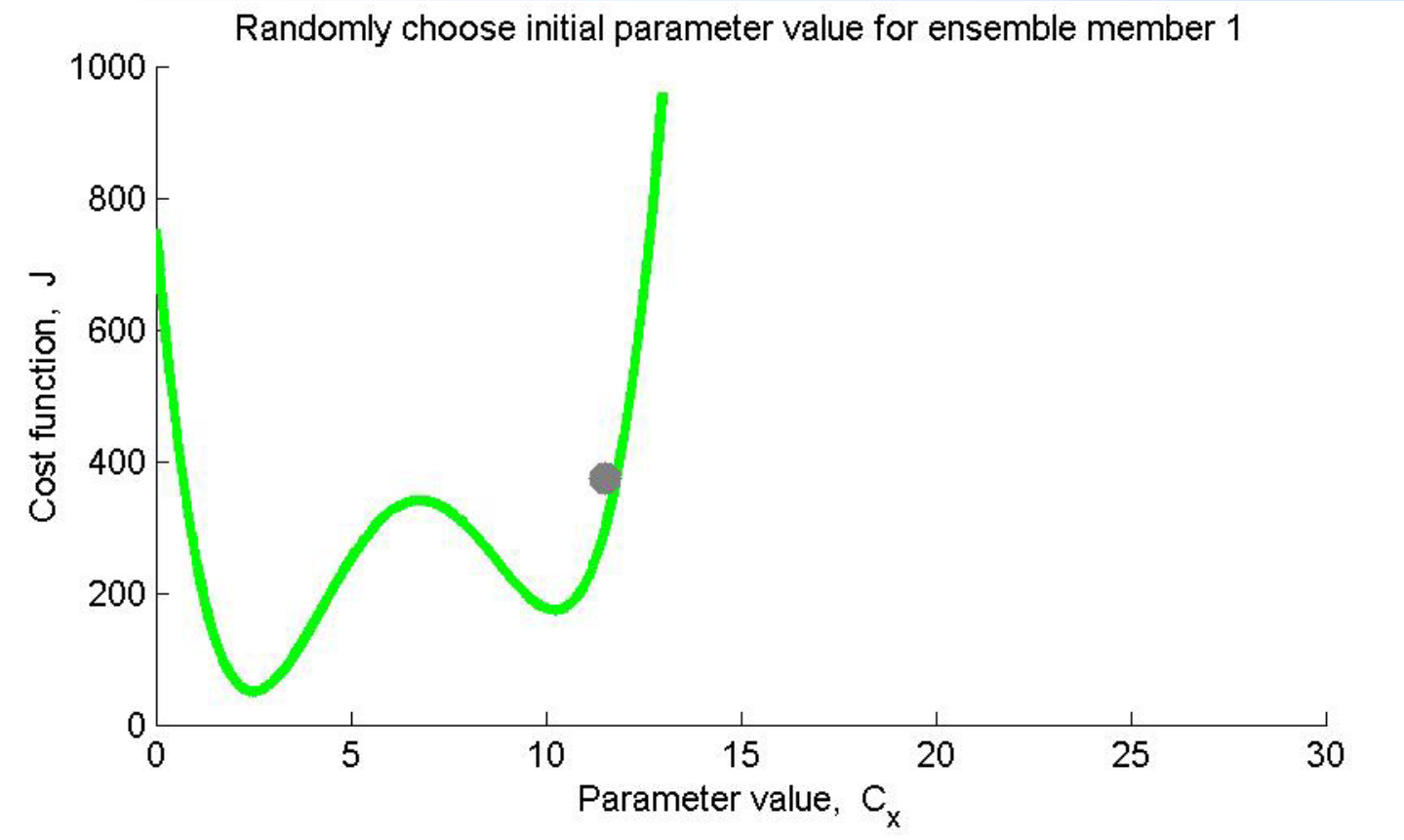
Each ensemble member from the Cu set is fit to the same data profile, but using different initial parameter values. Likewise for the Sc set.

Let’s illustrate the process schematically, assuming that we calibrate only one parameter

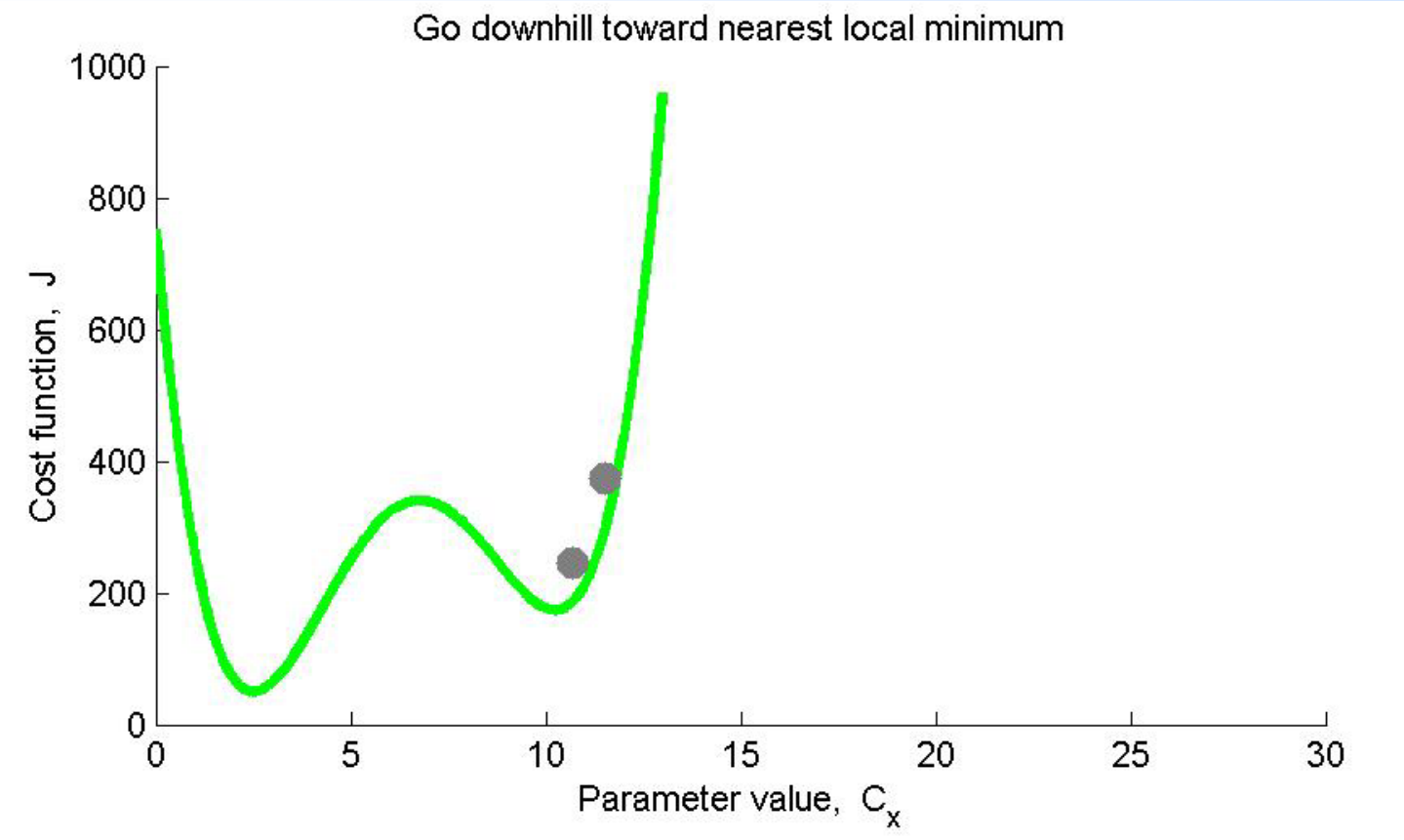
First calibrate to Cu data. Suppose the cost function has two minima



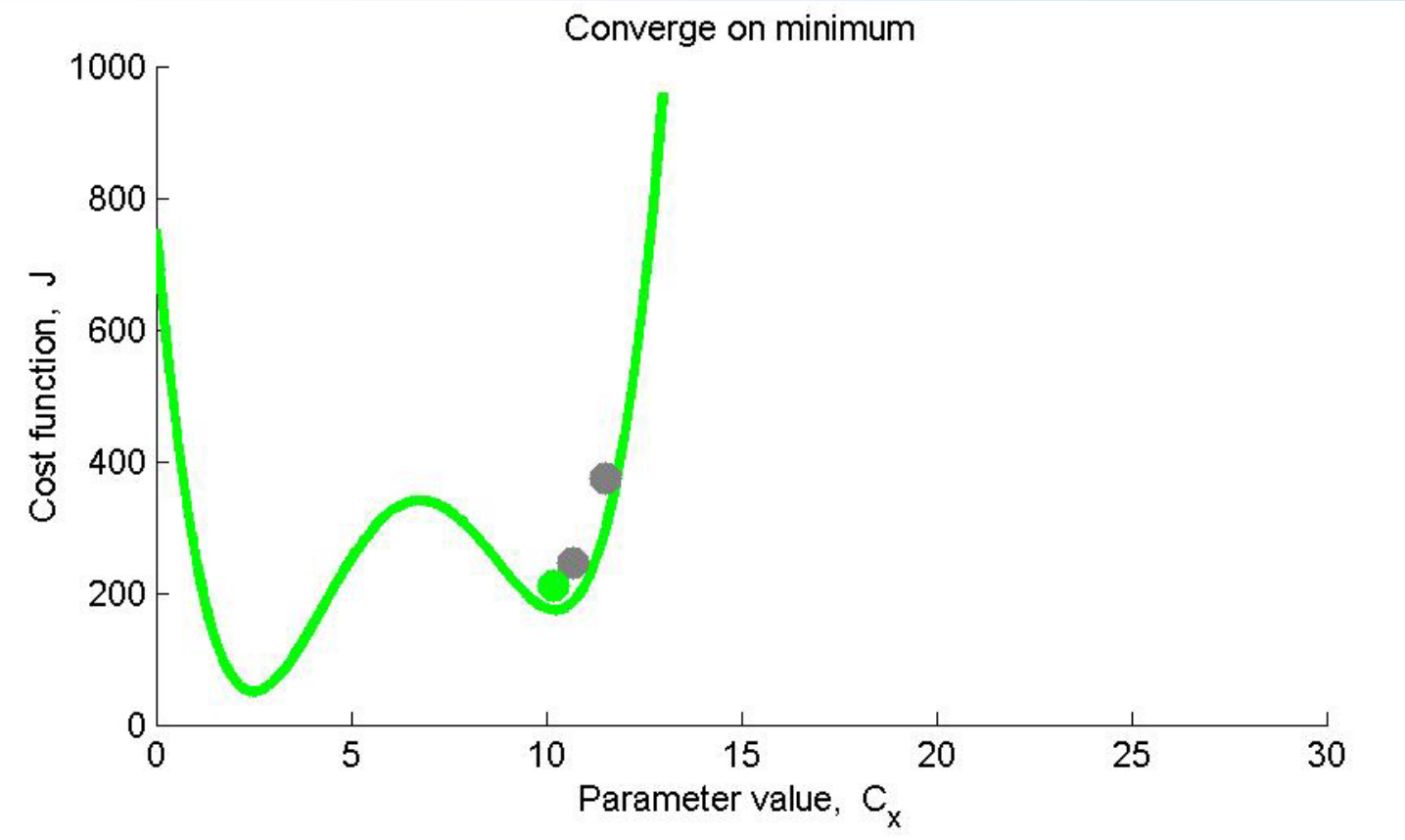
Choose C_x randomly, run the SCM, and compute cost function



Use the optimizer to choose an improved value of C_x, and then re-run the SCM

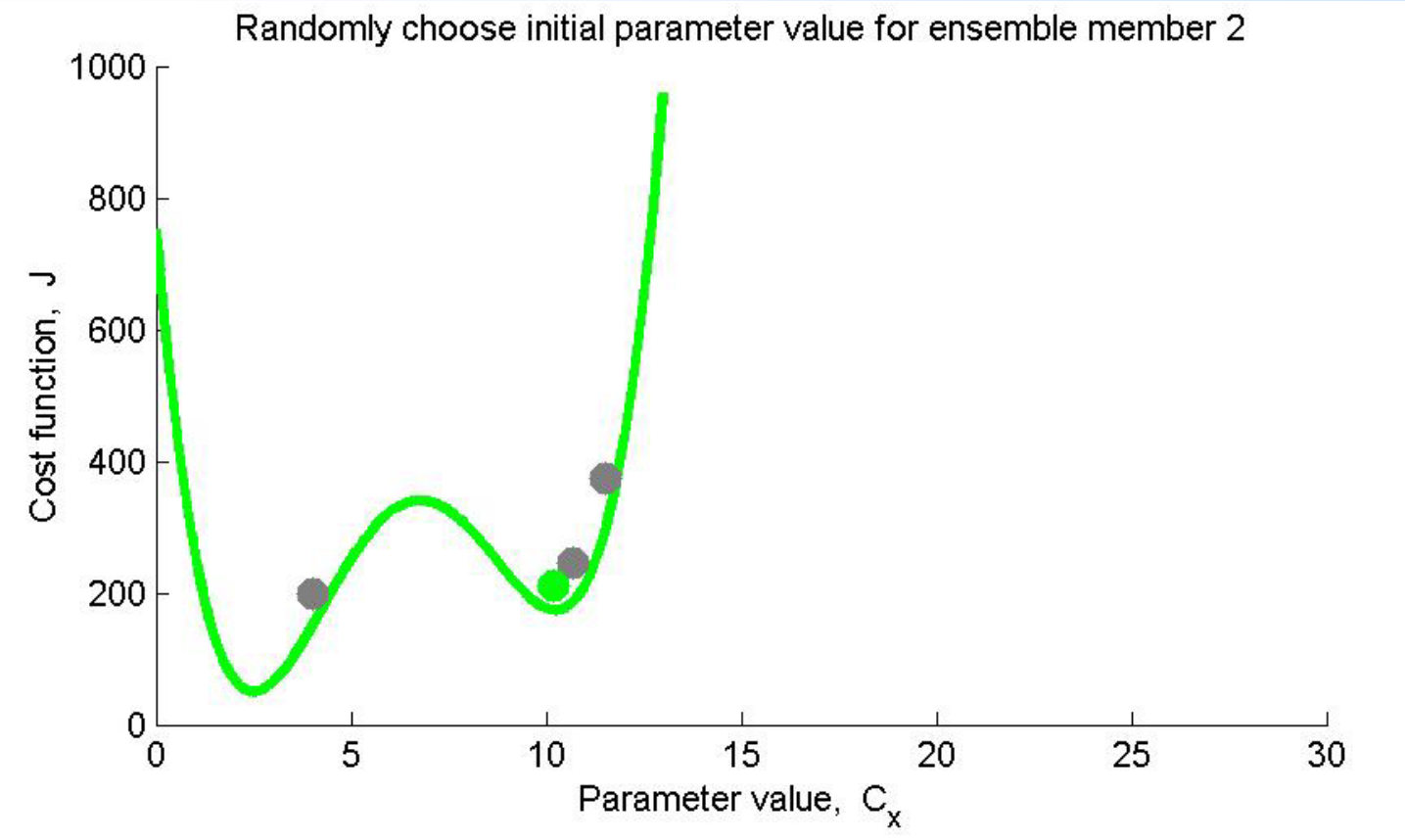


After multiple iterations, find the nearest local minimum.

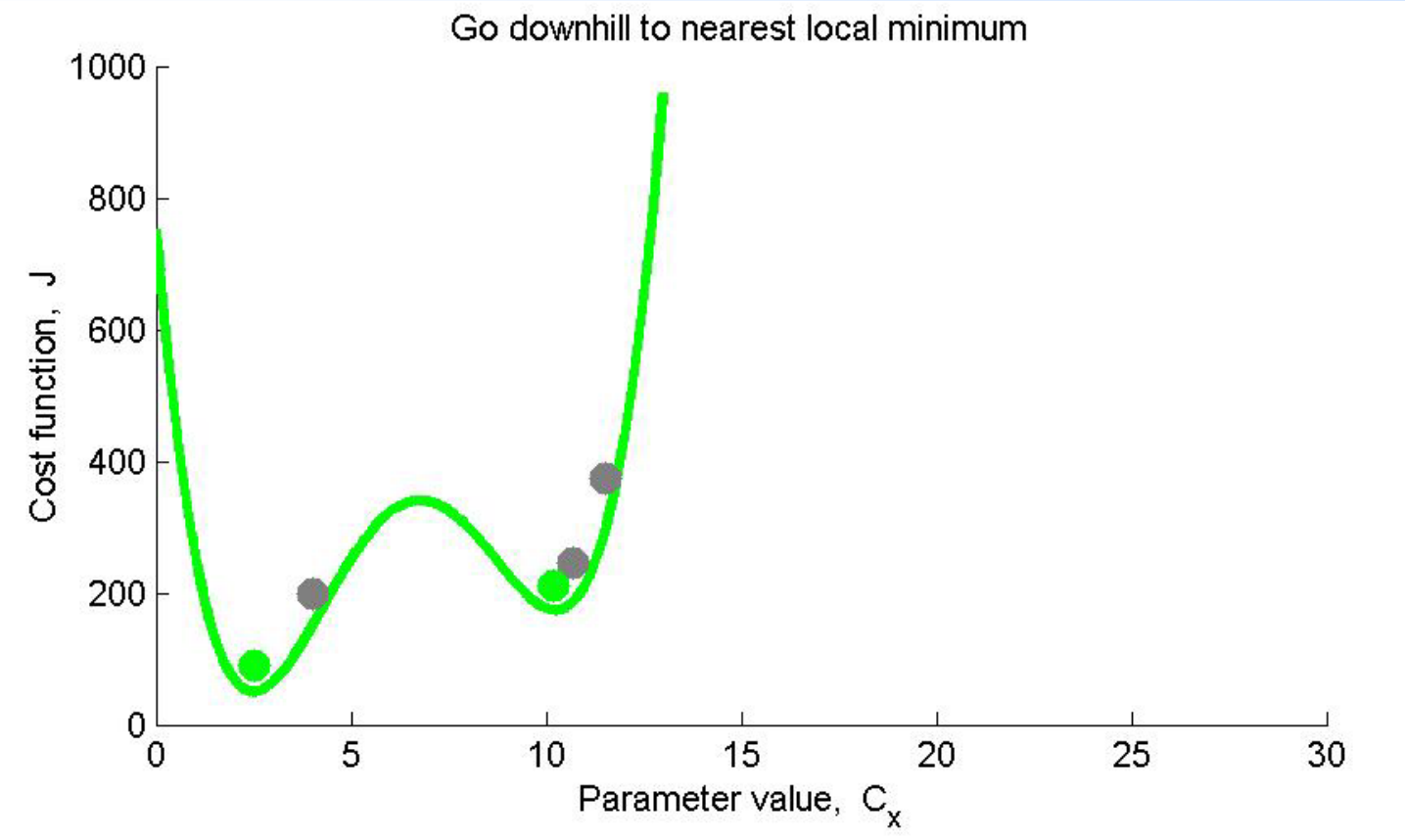


This is the value of C_x for the cumulus ensemble member 1.

Choose new value of C_x randomly

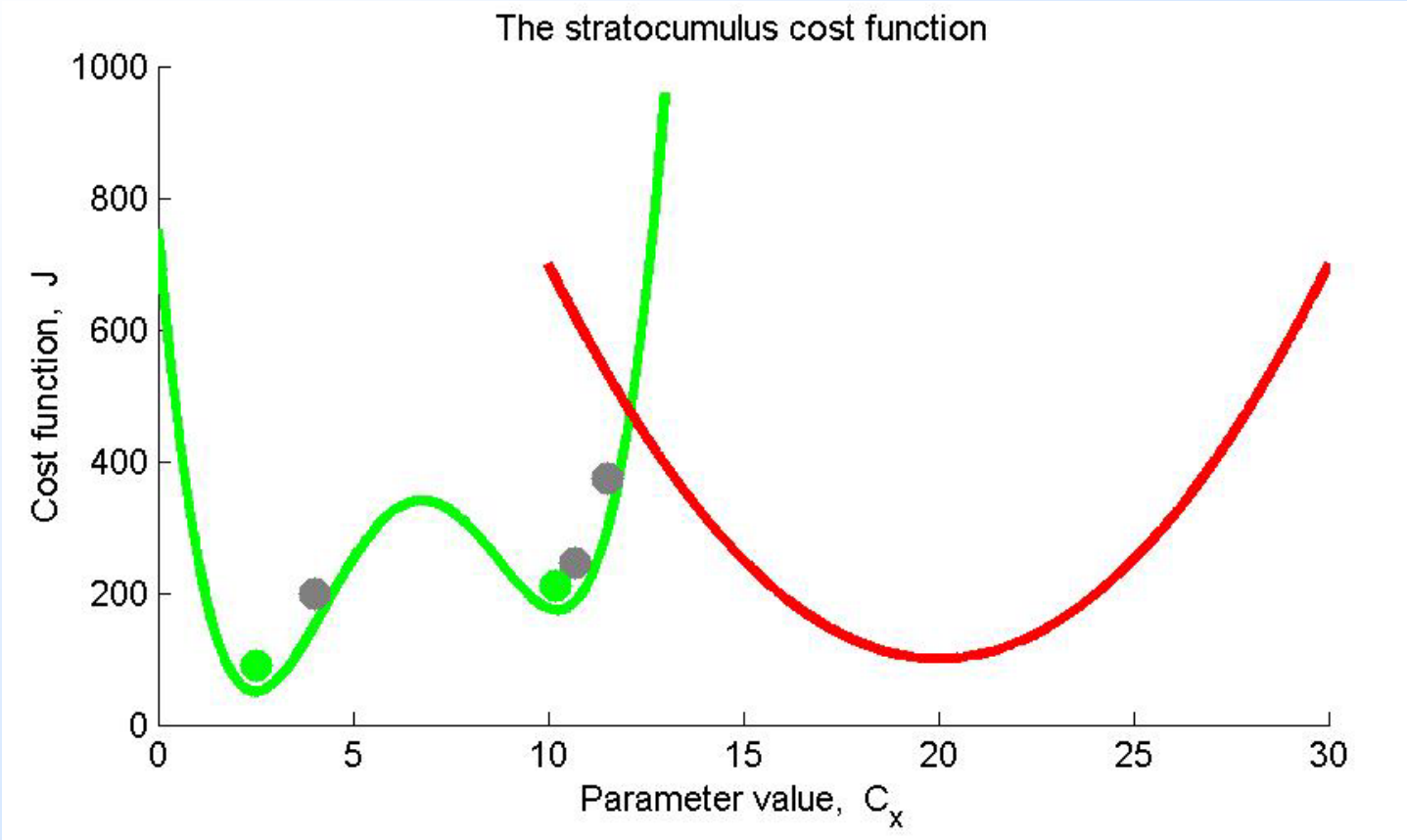


Find 2nd value of C_x



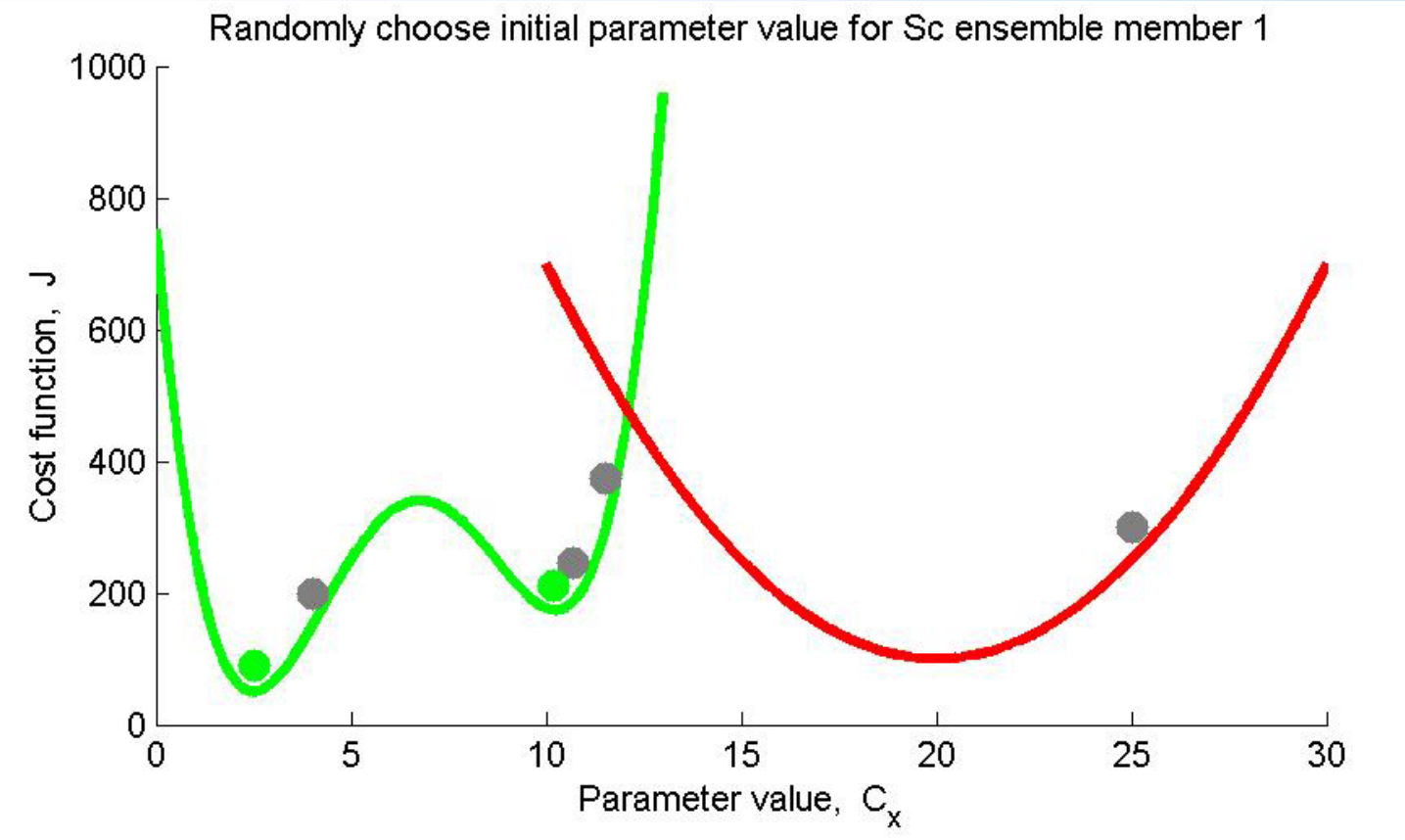
We have found the value of C_x for the cumulus ensemble member 2. The existence of multiple minima leads to a spread of "best-fit" parameter values.

Now calibrate separately to a Sc case

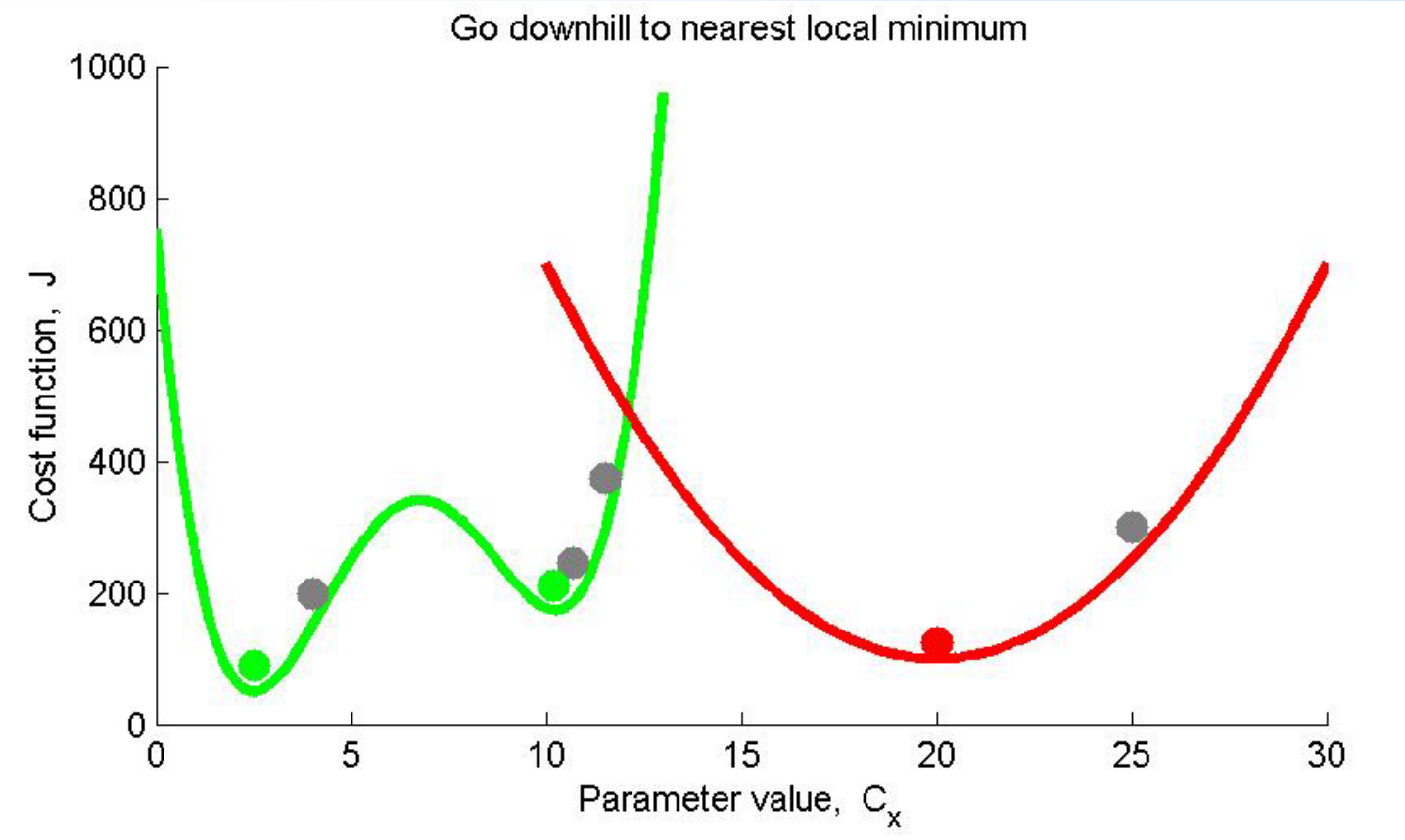


The Sc cost function is different because Sc is a fundamentally different cloud type than Cu.

Choose value of C_x randomly



Find value of C_x



The distributions of C_x may be different for Cu and Sc.

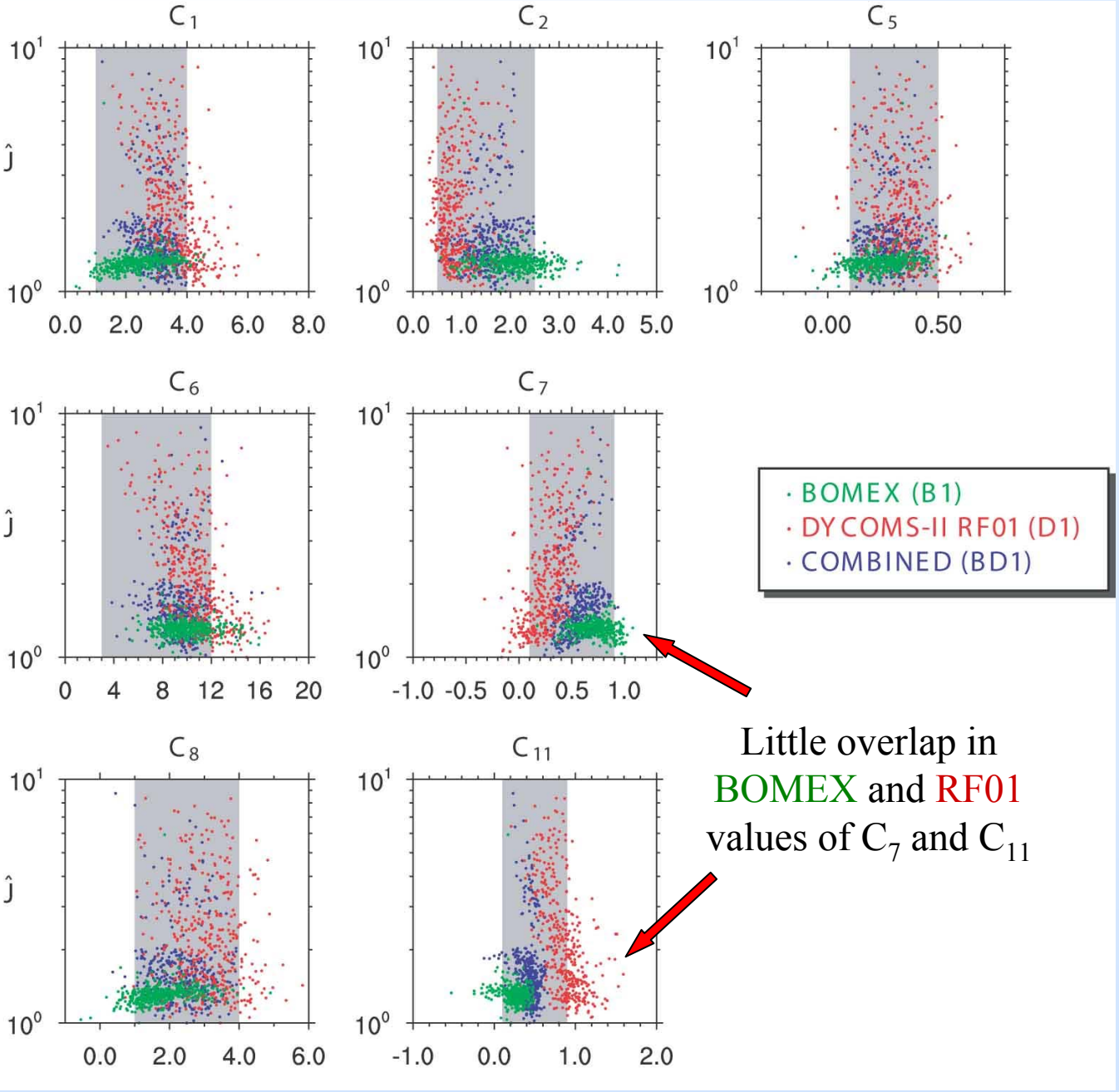
Outline

- Overview: Diagnosing model errors using calibration.
- Description of our single-column model for boundary layer clouds and turbulence.
- Our calibration methodology
- An example: Diagnosing errors in our single-column model.
- Can any of this aid superparameterization?

Initially, why did we want to diagnose errors in our single-column model?

Our first simulations of the DYCOMS-II RF01 stratocumulus case were poor. We thought that we might need to change our turbulent length scale. Before embarking on this time-consuming project, we wanted to know whether the error was parametric or structural, and whether we would need to change the length scale.

Therefore, we decided to calibrate our SCM.



Little overlap in BOMEX and RF01 values of C_7 and C_{11}

To diagnose model errors, we separately calibrate a trade-wind cumulus case (BOMEX) , a stratocumulus case (RF01) , and both datasets combined.

Each point represents a single calibration experiment, i.e. a single local minimum.

What have we learned from calibration about error in our SCM?

The gaps in C_7 and C_{11} parameter values indicates that the error in our SCM is not merely parametric, but structural. Namely, our SCM is underfit.

We should seek the source of this structural error, and not merely be content with the combined best-fit (blue) points.

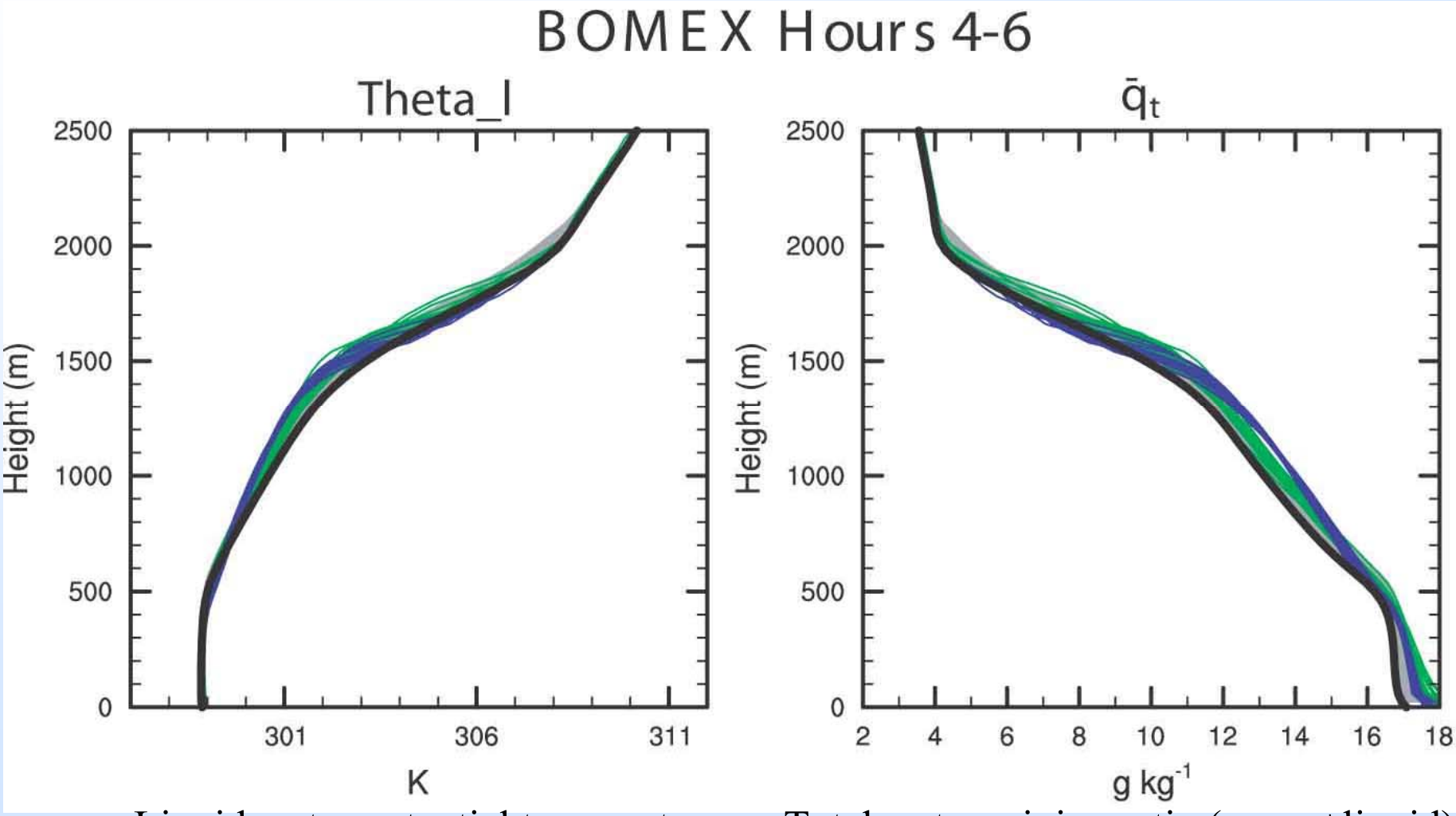
Caveats: Calibration is not a silver bullet

(1) Calibration locates the general area of the model error (it is related to C_7 and C_{11}), but does not pinpoint its exact source.

(2) Calibration does not propose a new model structure. For that, we need to be creative.

What do the calibrated profiles look like?

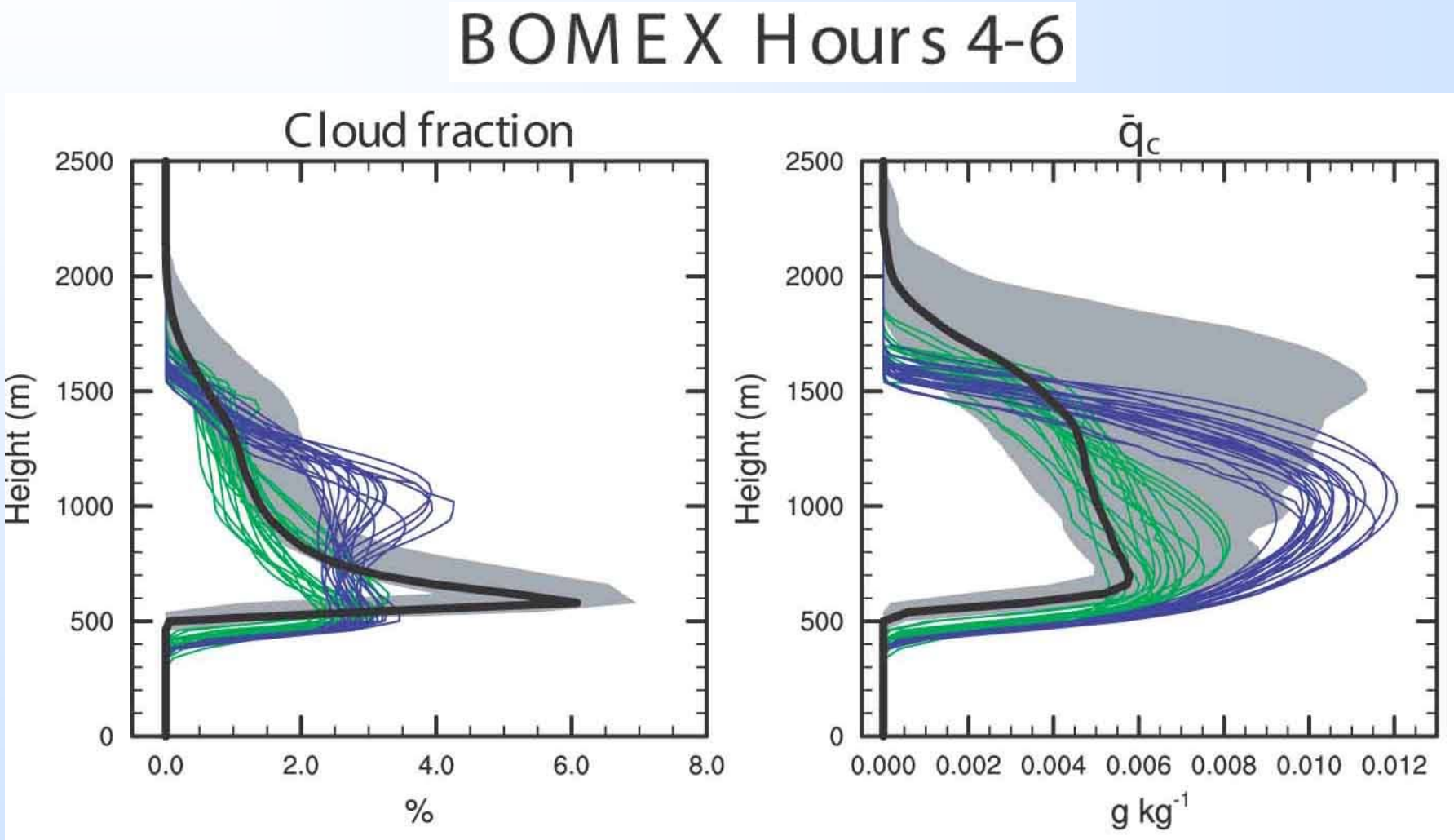
A good fit to both BOMEX Cu-only and combined datasets. (Thick black line is LES)



Liquid water potential temperature Total water mixing ratio (vapor+liquid)

E.g., each green line corresponds to one ensemble member, i.e. one green dot.

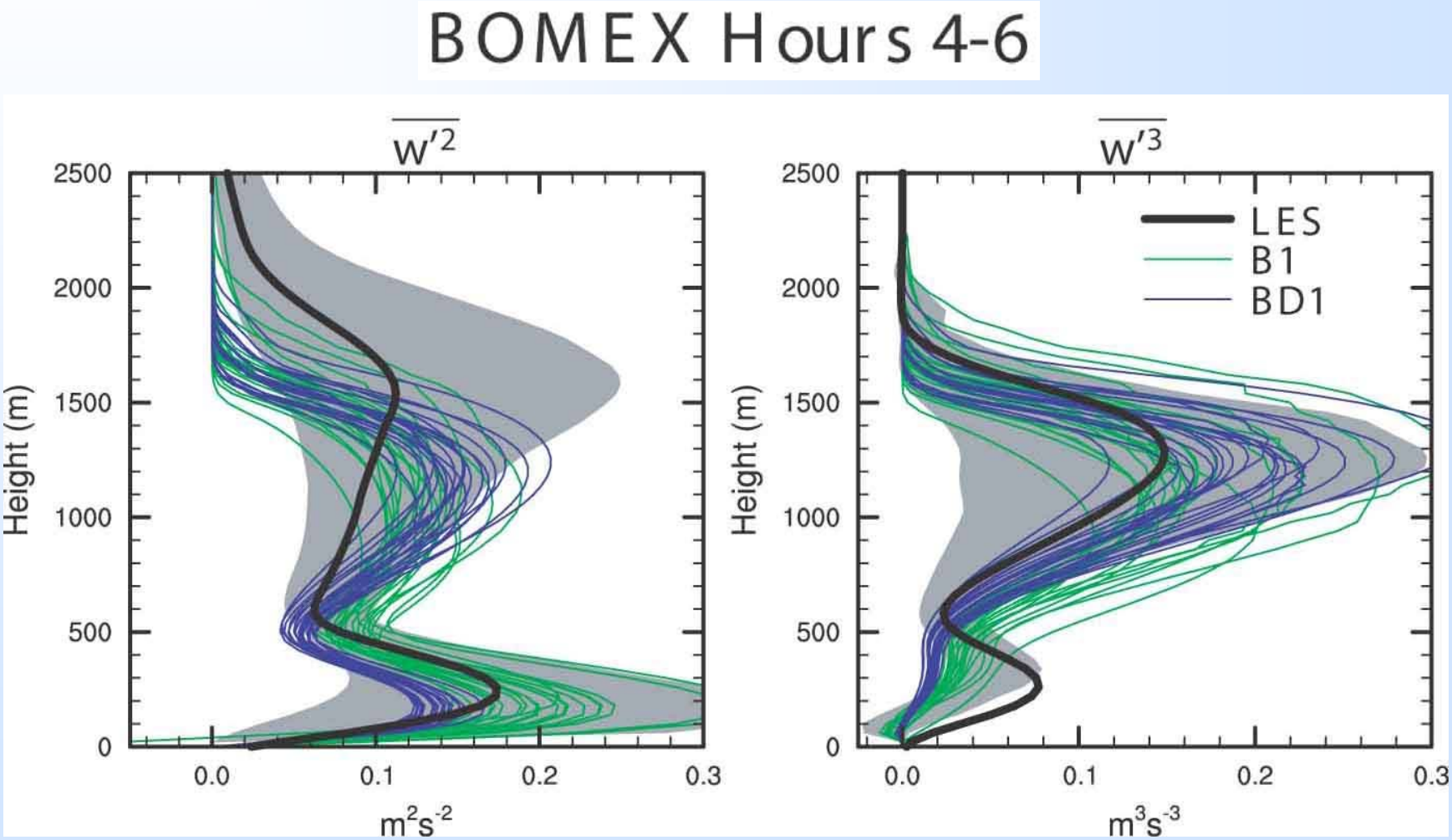
A problematic cloud fraction fit to the combined dataset



Cloud fraction

Liquid water mixing ratio

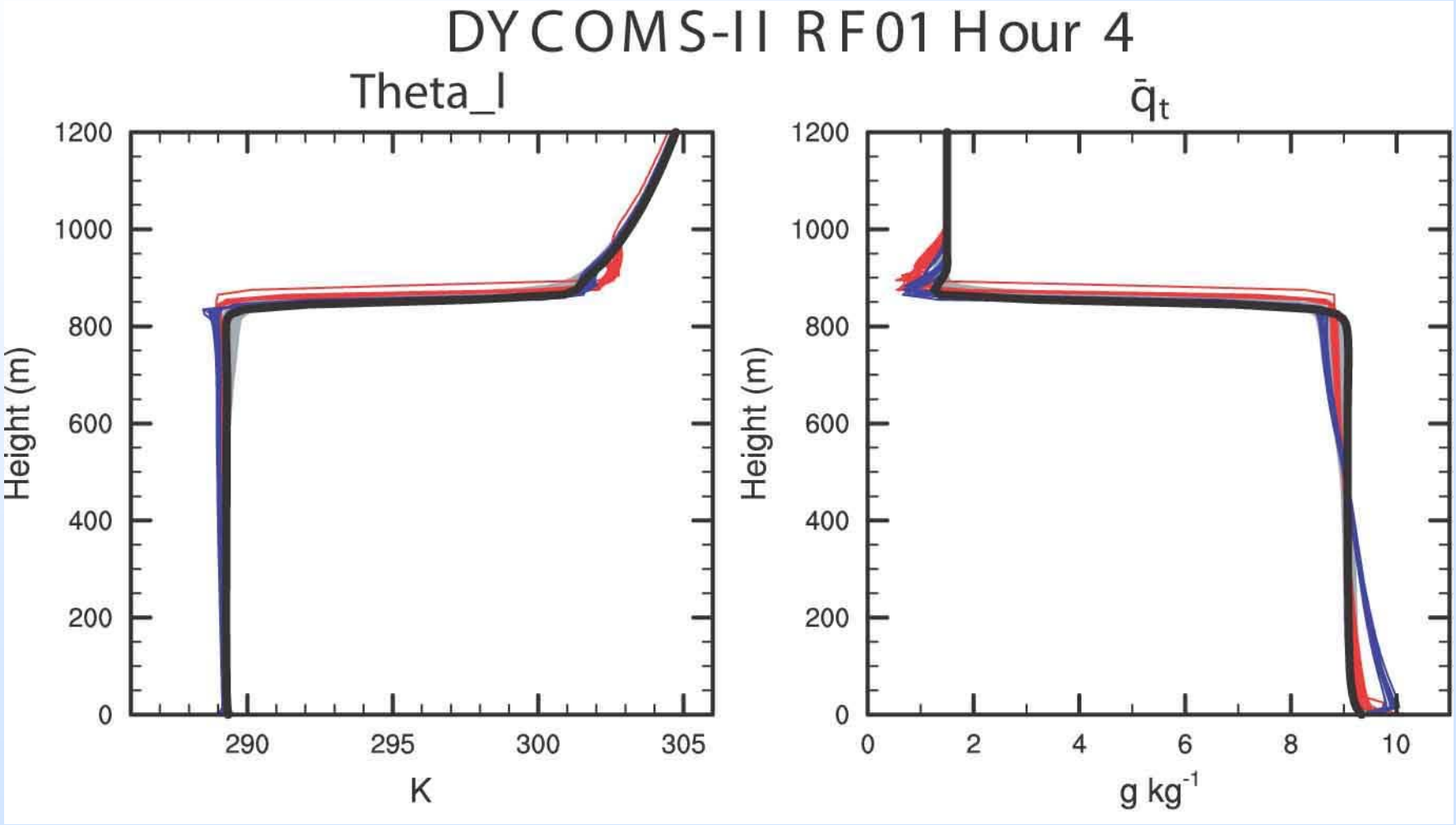
A good fit to both BOMEX Cu-only and combined datasets



Vertical velocity variance

Vertical velocity third moment

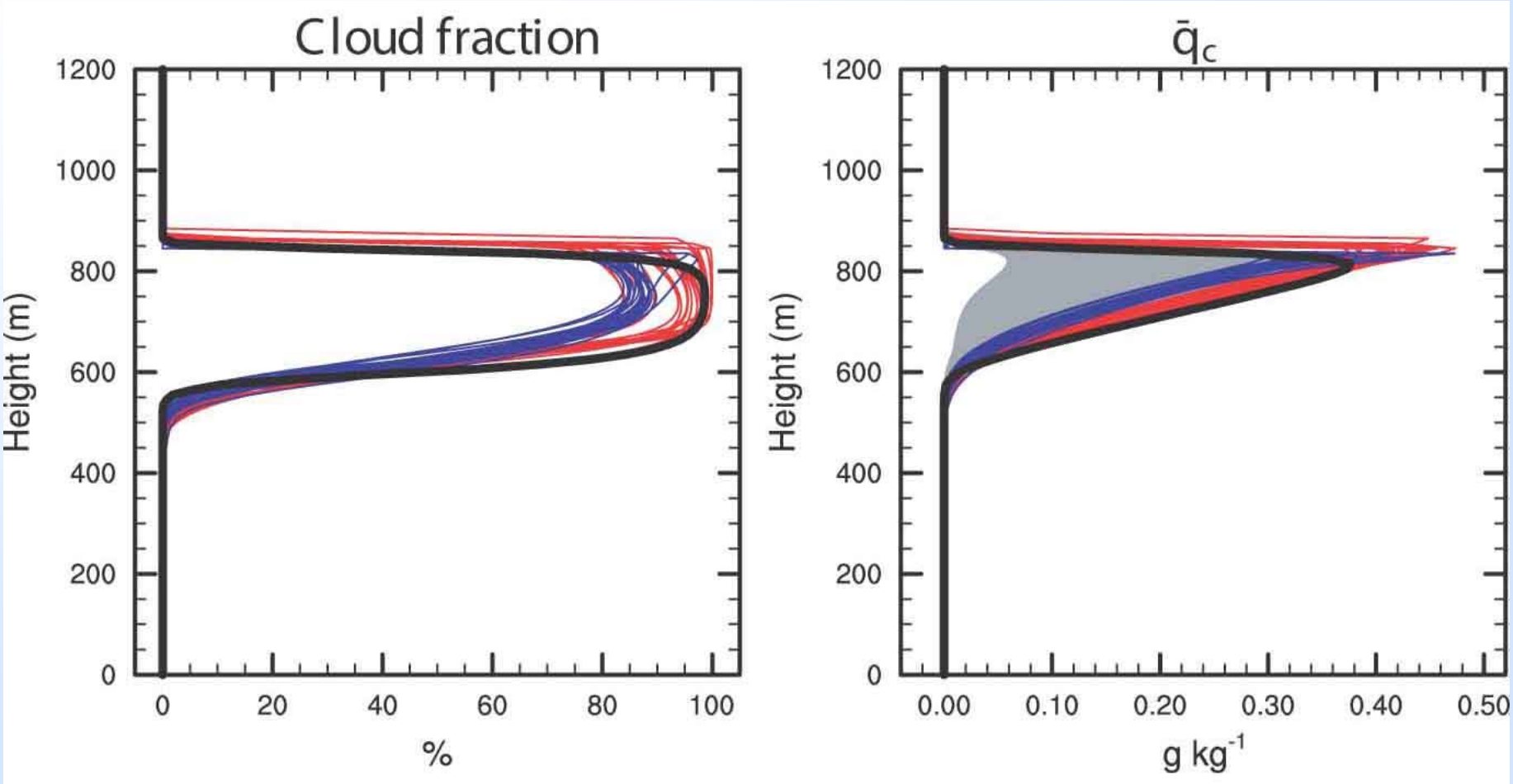
A non-well-mixed total water fit to the combined datasets



Liquid water potential temperature

Total water mixing ratio

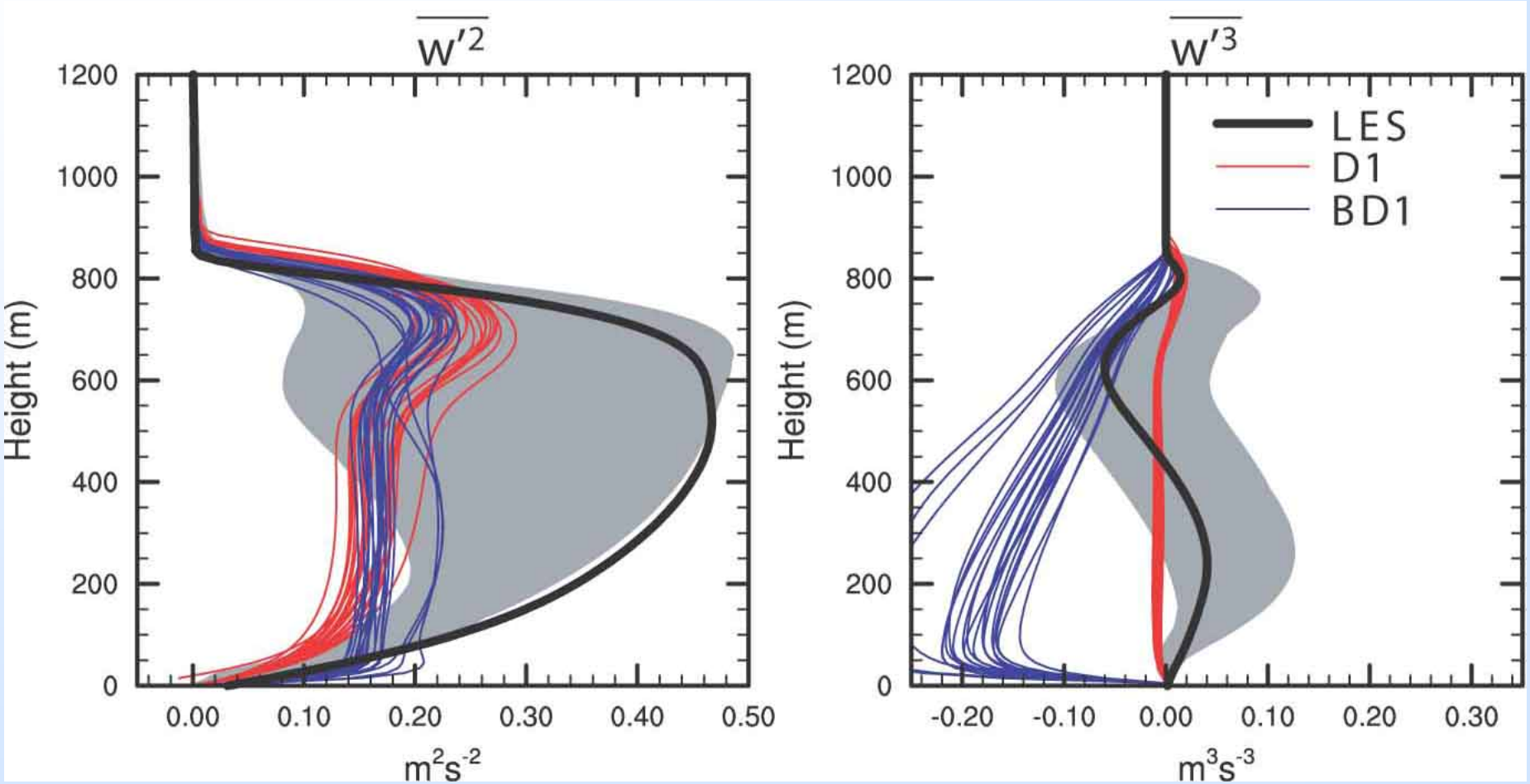
A good fit to both RF01 Sc-only and combined datasets



Cloud fraction

Liquid water mixing ratio

A poor fit of w'^3 to the combined dataset



Vertical velocity variance

Third moment of vertical velocity

The two non-overlapping coefficients, C_7 and C_{11}, pre-multiply pressure terms

$$\left.\frac{\partial \overline{w'q_t'}}{\partial t}\right|_{\text{pressure}} = -\frac{1}{\rho_0}\overline{q_t'\frac{\partial p'}{\partial z}} = -\frac{C_6}{\tau}\overline{w'q_t'} - C_7\frac{g}{\theta_0}\overline{q_t'\theta_v'}$$

$$\left.\frac{\partial \overline{w'^3}}{\partial t}\right|_{\text{pressure}} = -\frac{3}{\rho_0}\overline{w'^2\frac{\partial p'}{\partial z}}$$

$$= -\frac{C_8}{\tau}\left(C_{8b}Sk_w^4 + 1\right)\overline{w'^3} - C_{11}\frac{3g}{\theta_0}\overline{w'^2\theta_v'}$$

θ_v = virtual potential temperature

Physical explanation of the C_{11} term

Using our SCM, the Cu simulations were too Sc-like; and the Sc simulations were too Cu-like.

A key difference between Cu and Sc layers: Cu have large positive skewness, and Sc have skewness of less magnitude.

The C_{11} term damps skewness.

Our SCM was damping skewness too much in Cu layers, and too little in Sc layers.

We address the underfitting by turning the constants into functions

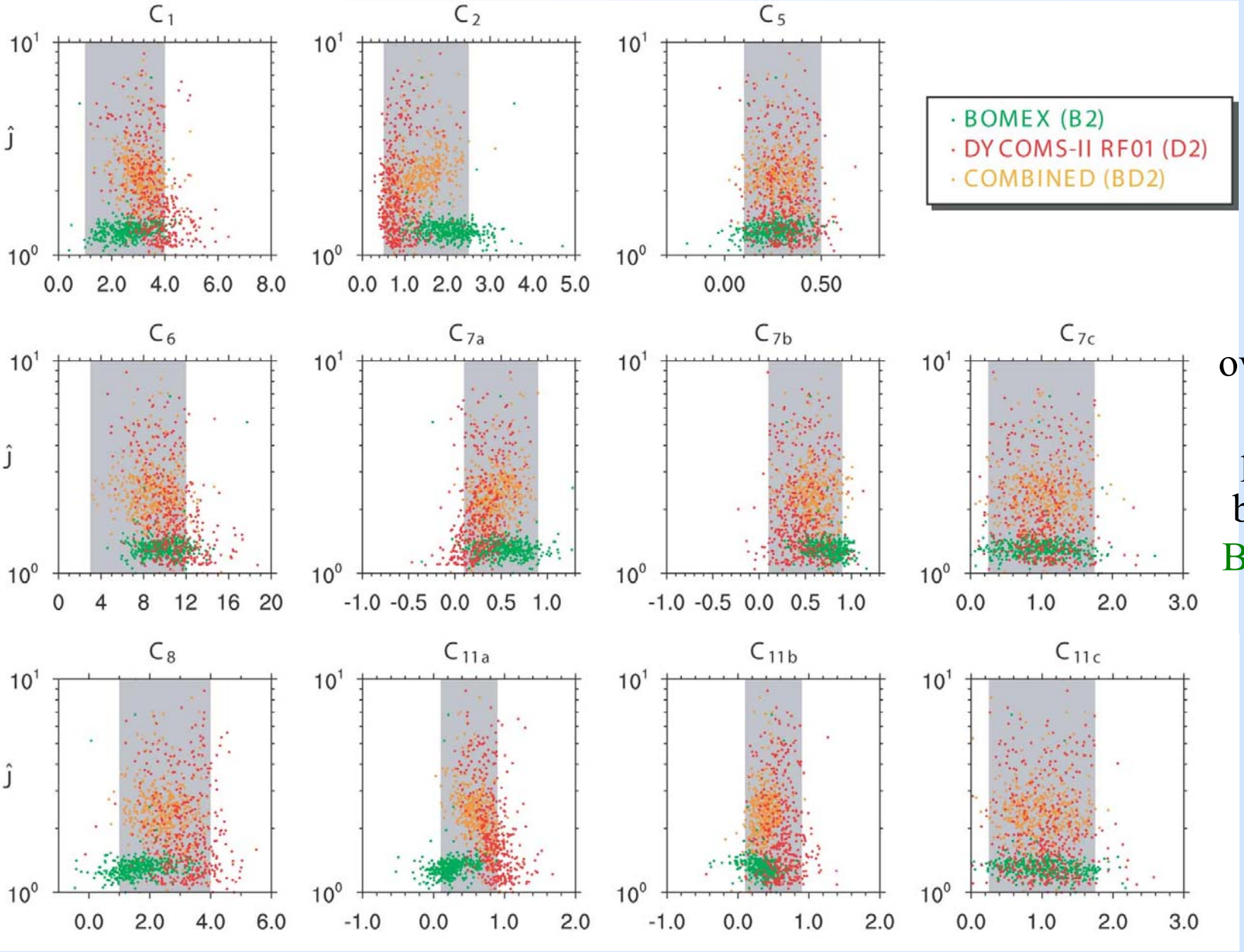
Cumulus have high skewness; stratocumulus have low skewness. We try the following empirical modification:

$$C_7(Sk_w) = C_{7b} + (C_{7a} - C_{7b})\, e^{-\frac{1}{2}\left(\frac{Sk_w}{C_{7c}}\right)^2}$$

$$C_{11}(Sk_w) = C_{11b} + (C_{11a} - C_{11b})\, e^{-\frac{1}{2}\left(\frac{Sk_w}{C_{11c}}\right)^2}$$

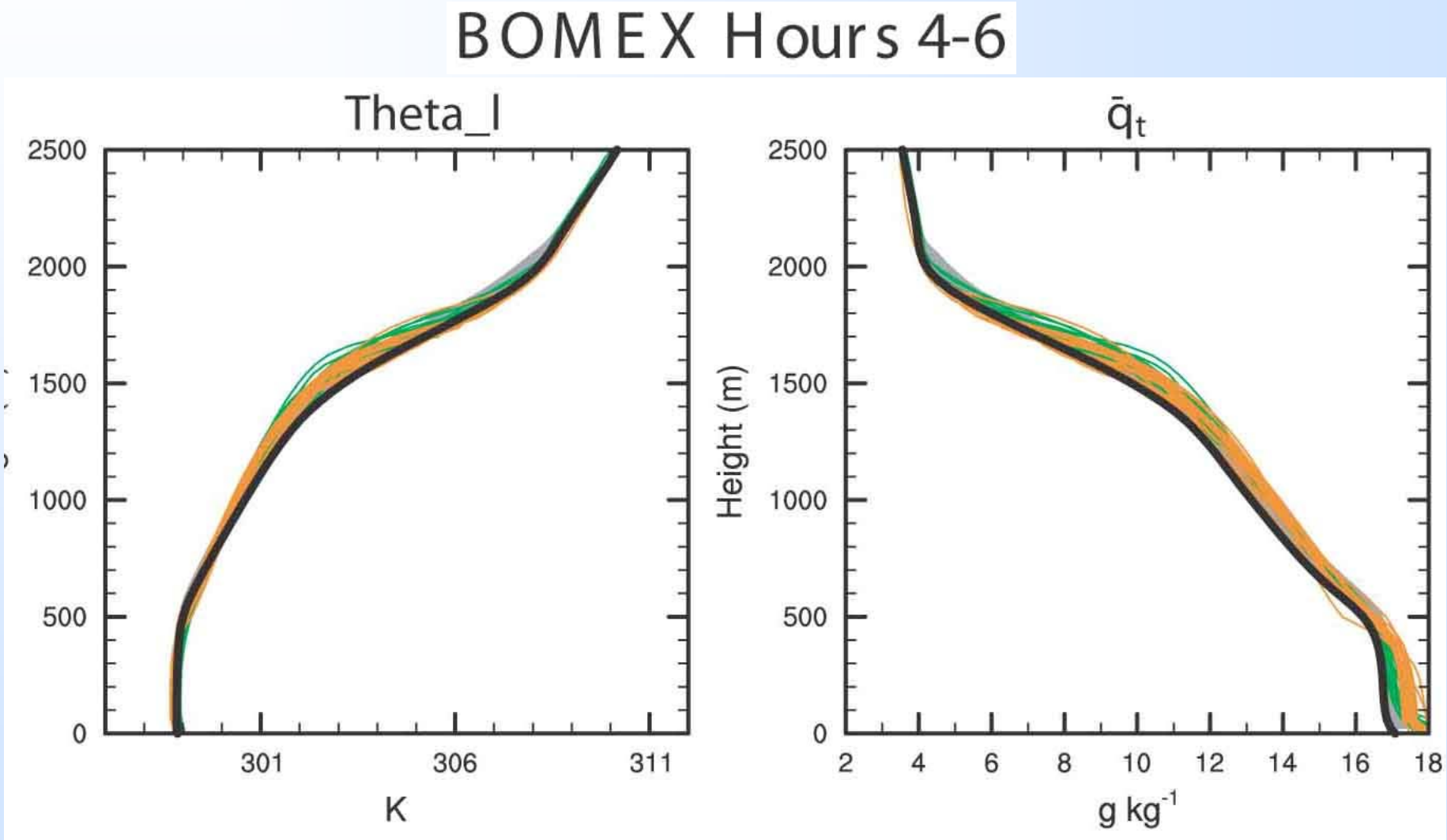
Here Sk_w is the skewness of vertical velocity. Although this is empirical and preliminary, at least it allows us to test the hypothesis that skewness is important in an interactive simulation.

We now re-calibrate the revised model with skewness-dependent functions for C_7 and C_{11}



There is improved overlap in the C_7 and C_{11} parameters between the BOMEX and RF01 datasets.

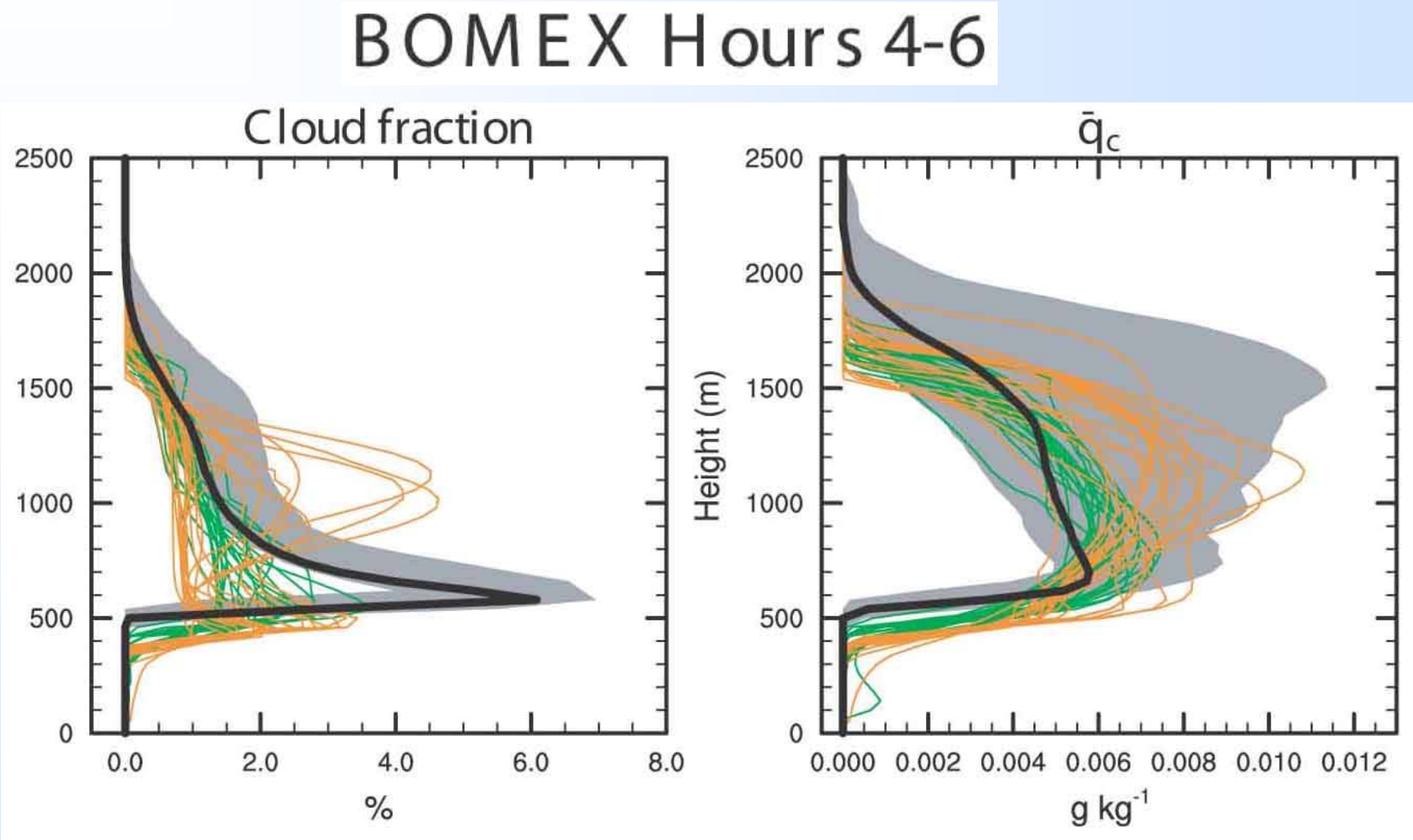
A good fit to both BOMEX Cu-only and combined datasets



Liquid water potential temperature

Total water mixing ratio

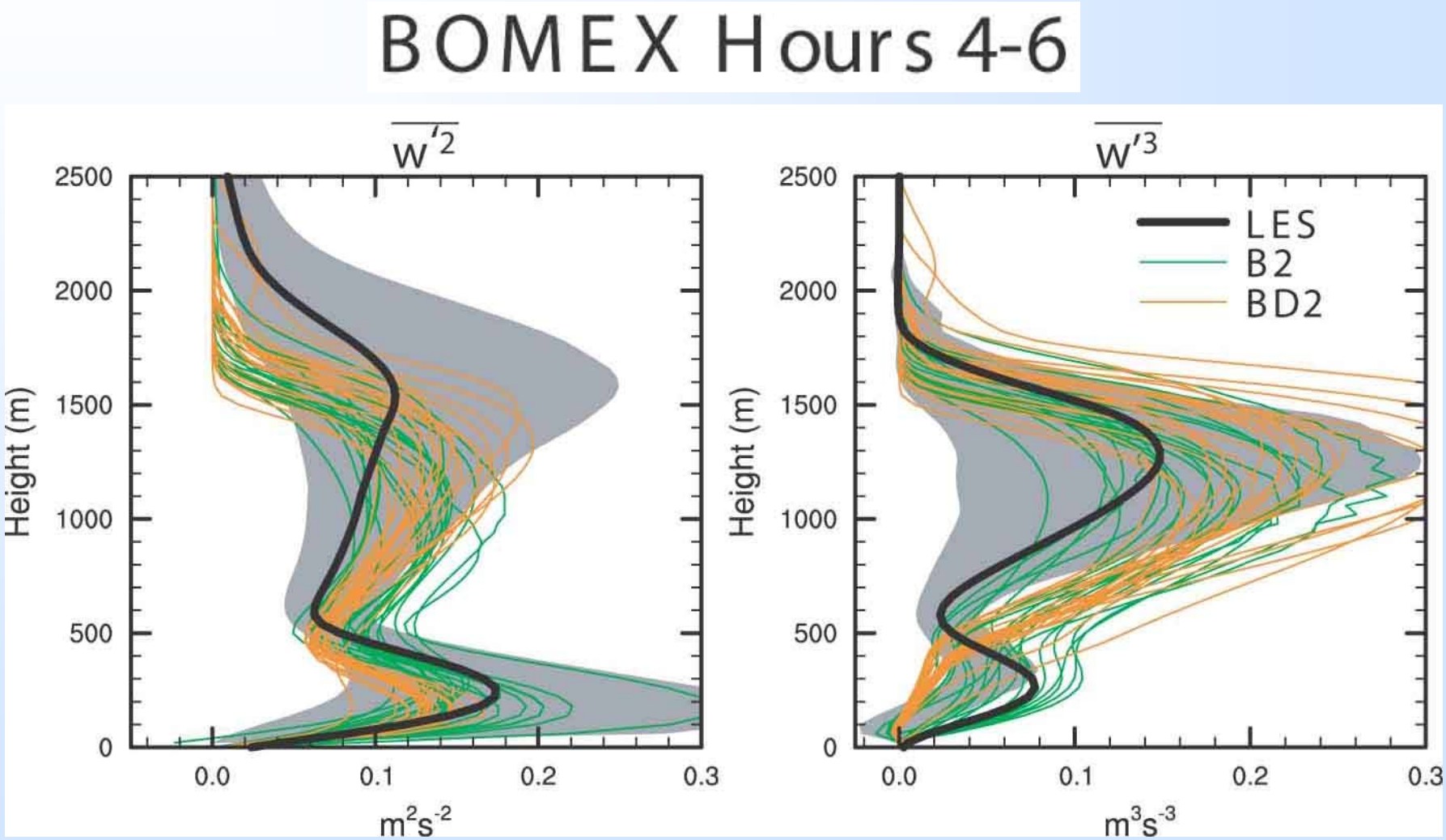
A somewhat improved fit to the combined dataset



Cloud fraction

Liquid water mixing ratio

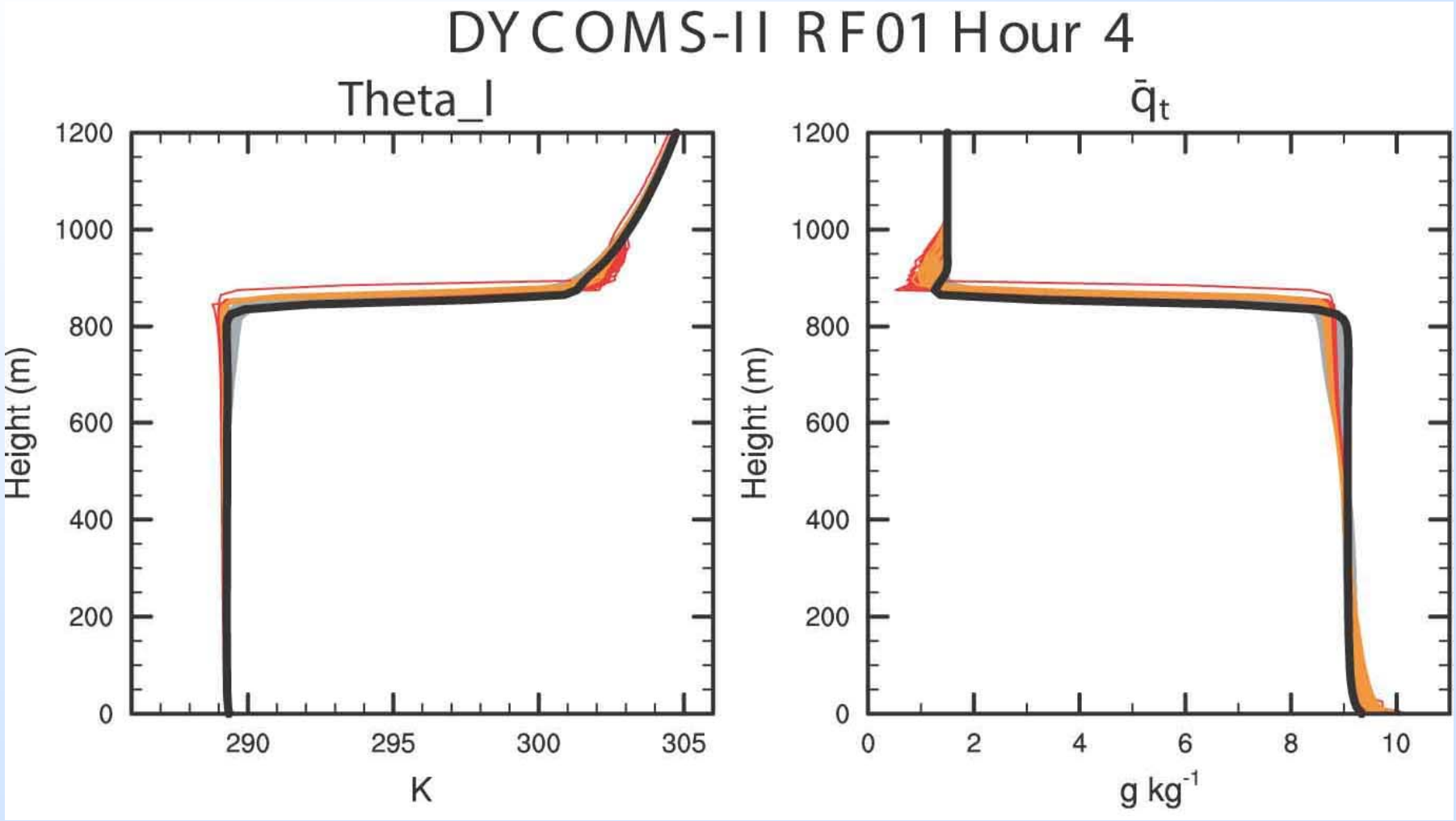
A good fit to both BOMEX Cu-only and combined datasets



Vertical velocity variance

Third moment of vertical velocity

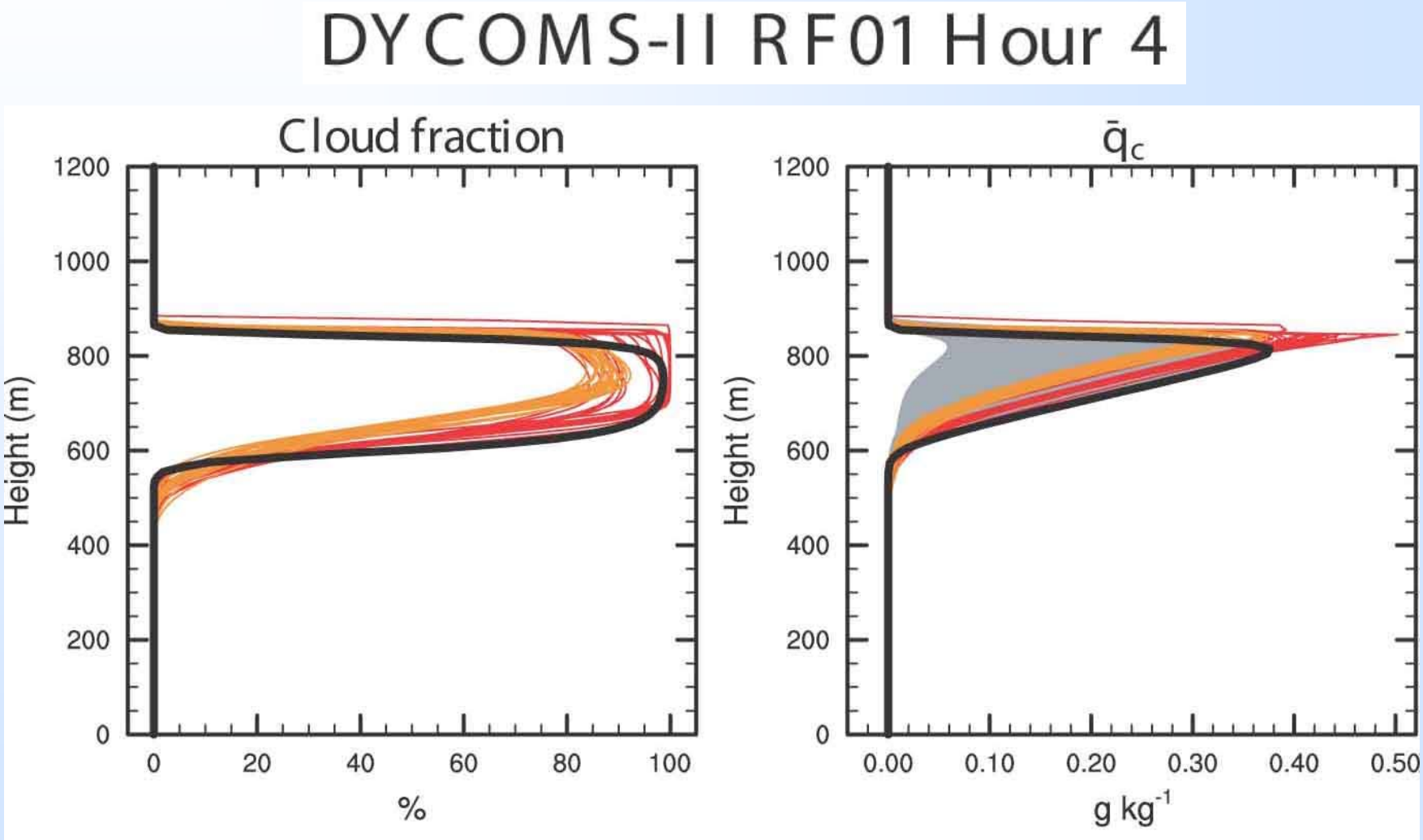
A more well-mixed total water for the combined dataset



Liquid water potential temperature

Total water mixing ratio

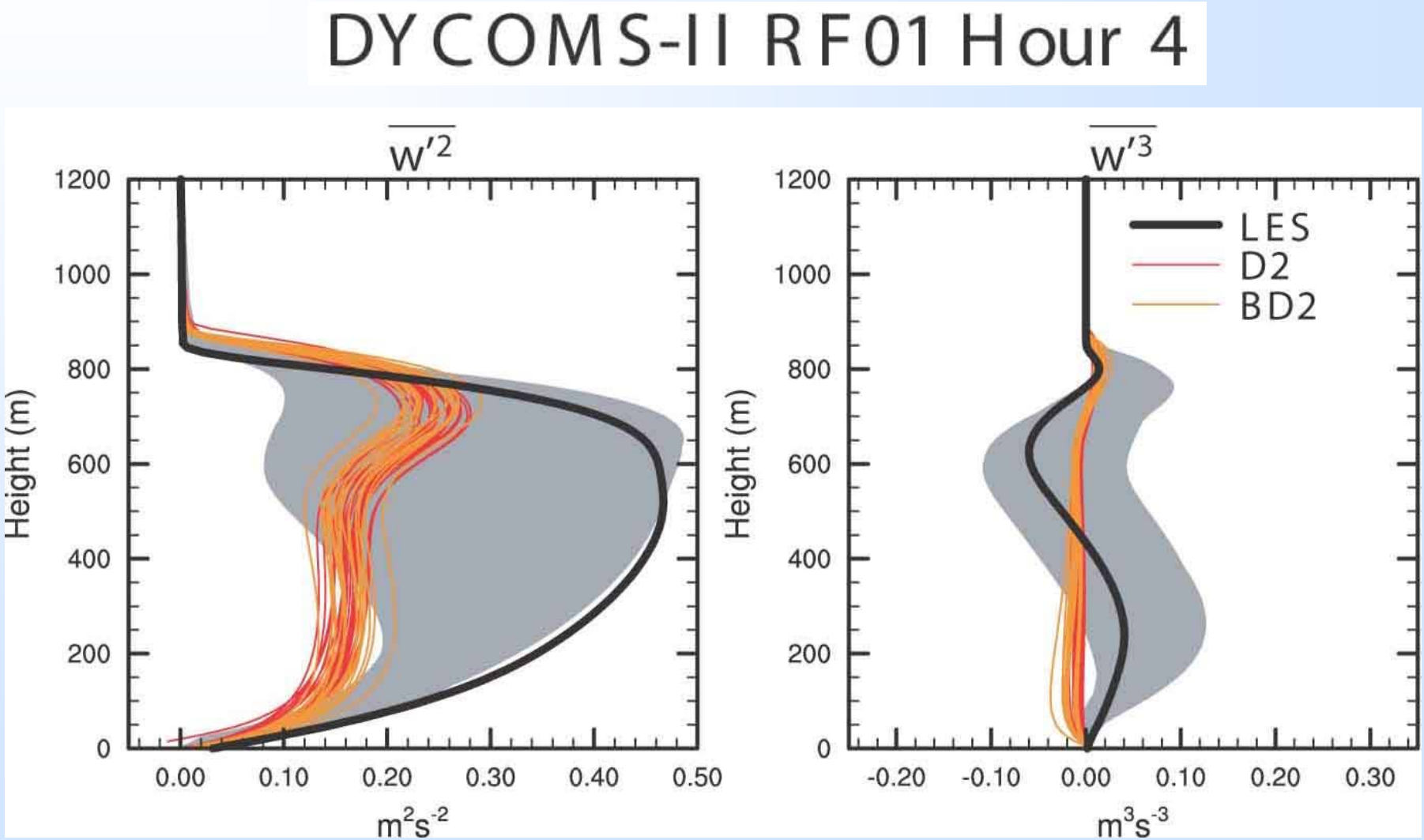
A good fit to both RF01 Sc-only and combined datasets



Cloud fraction

Liquid water mixing ratio

A reasonable fit to both RF01 Sc-only and combined datasets



Vertical velocity variance

Third moment of vertical velocity

Summary: the new skewness dependent parameters alleviate the underfitting to some extent

- BOMEX Cu liquid water is improved in some ensemble members

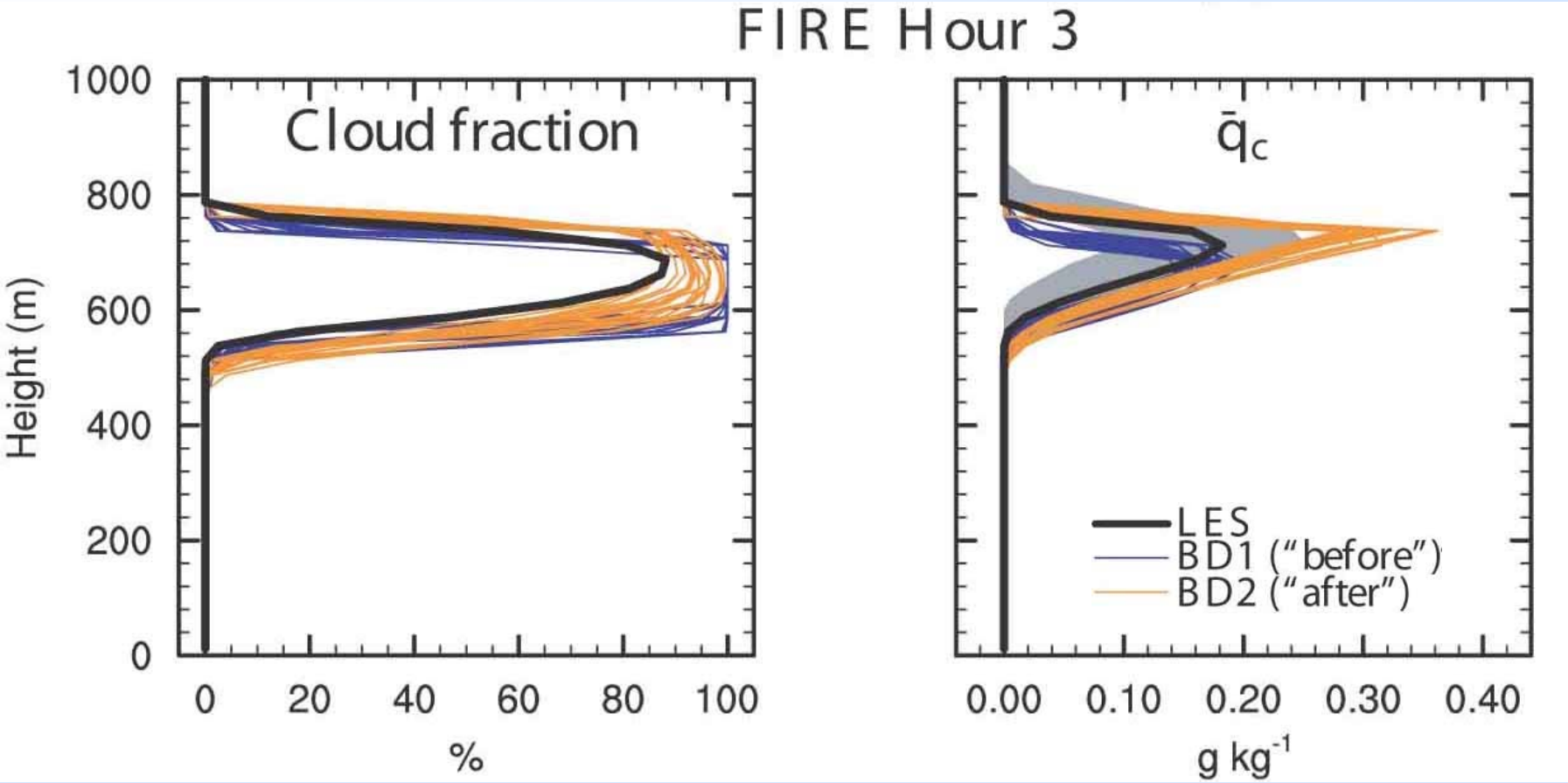
- RF01 Sc is more well mixed and has a more realistic skewness profile

This demonstrates that we can obtain improved fits without changing the length scale.

Now we check that the SCM is not overfit by cross-validation against other cloud cases

By improving the fit to the BOMEX Cu case and the RF01 Sc case, have we damaged the fit to other cases?

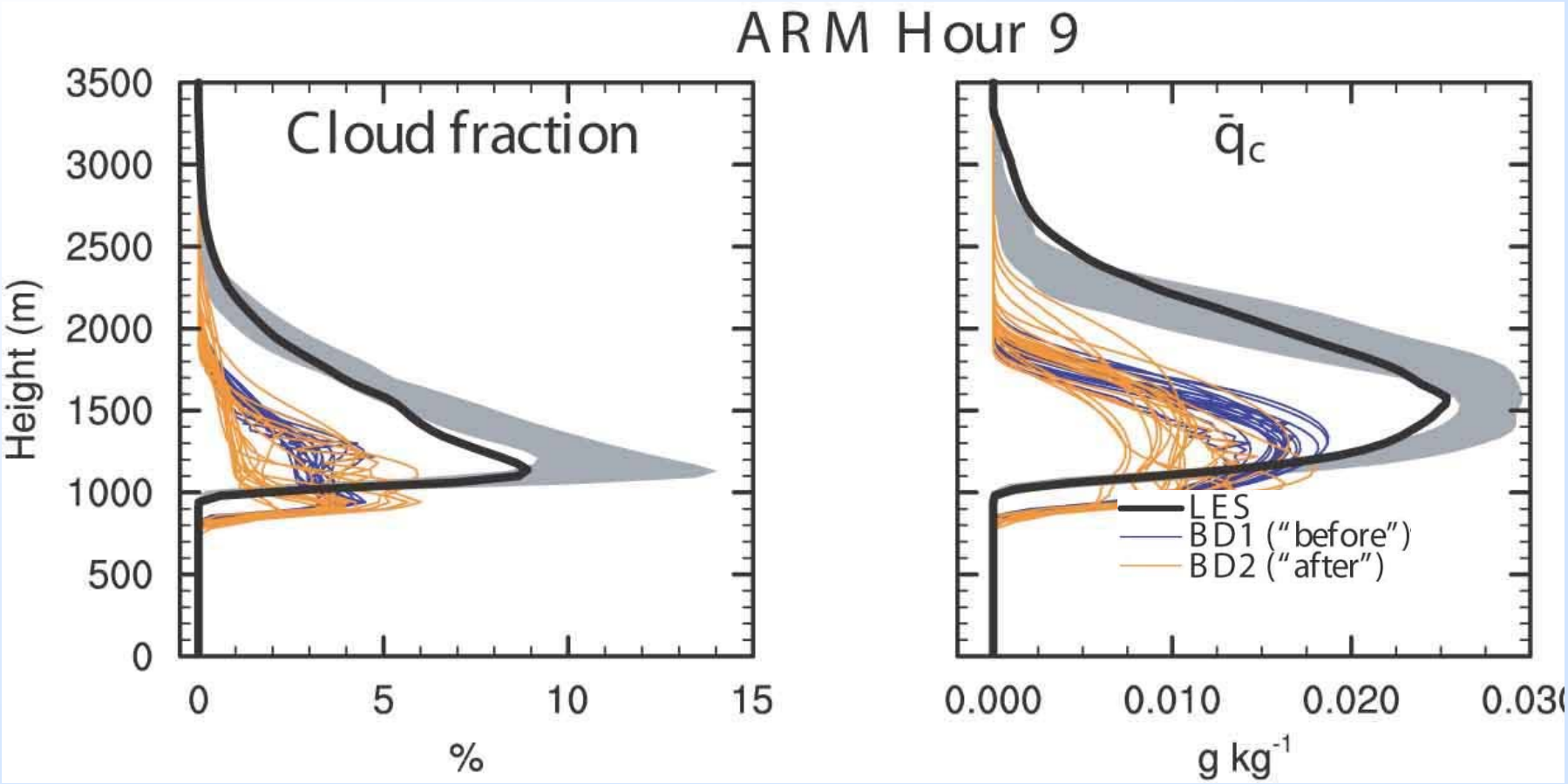
The FIRE marine stratocumulus case changes but doesn't improve or degrade



Cloud fraction

Liquid water mixing ratio

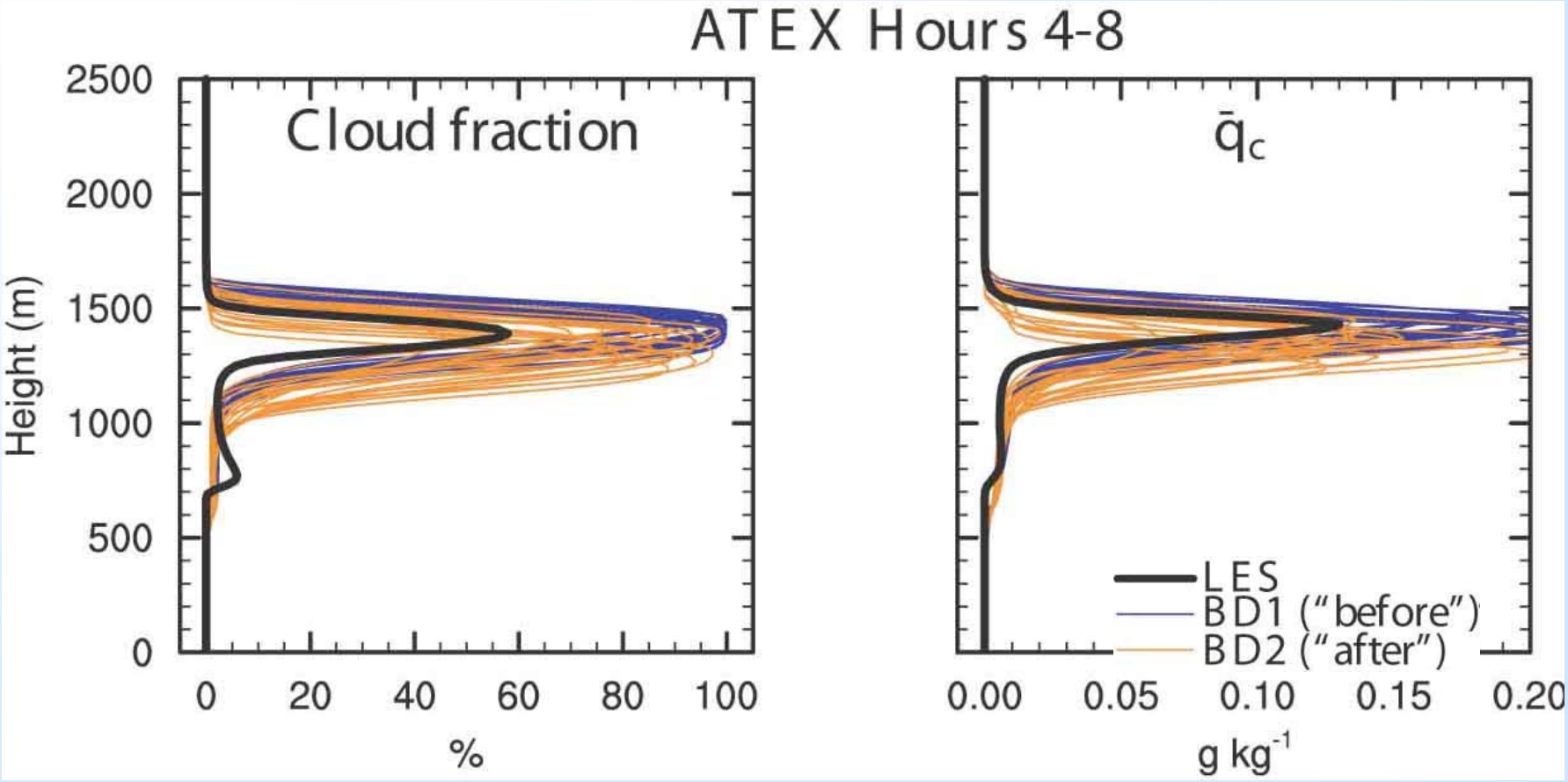
The ARM continental shallow cumulus case doesn't change much



Cloud fraction

Liquid water mixing ratio

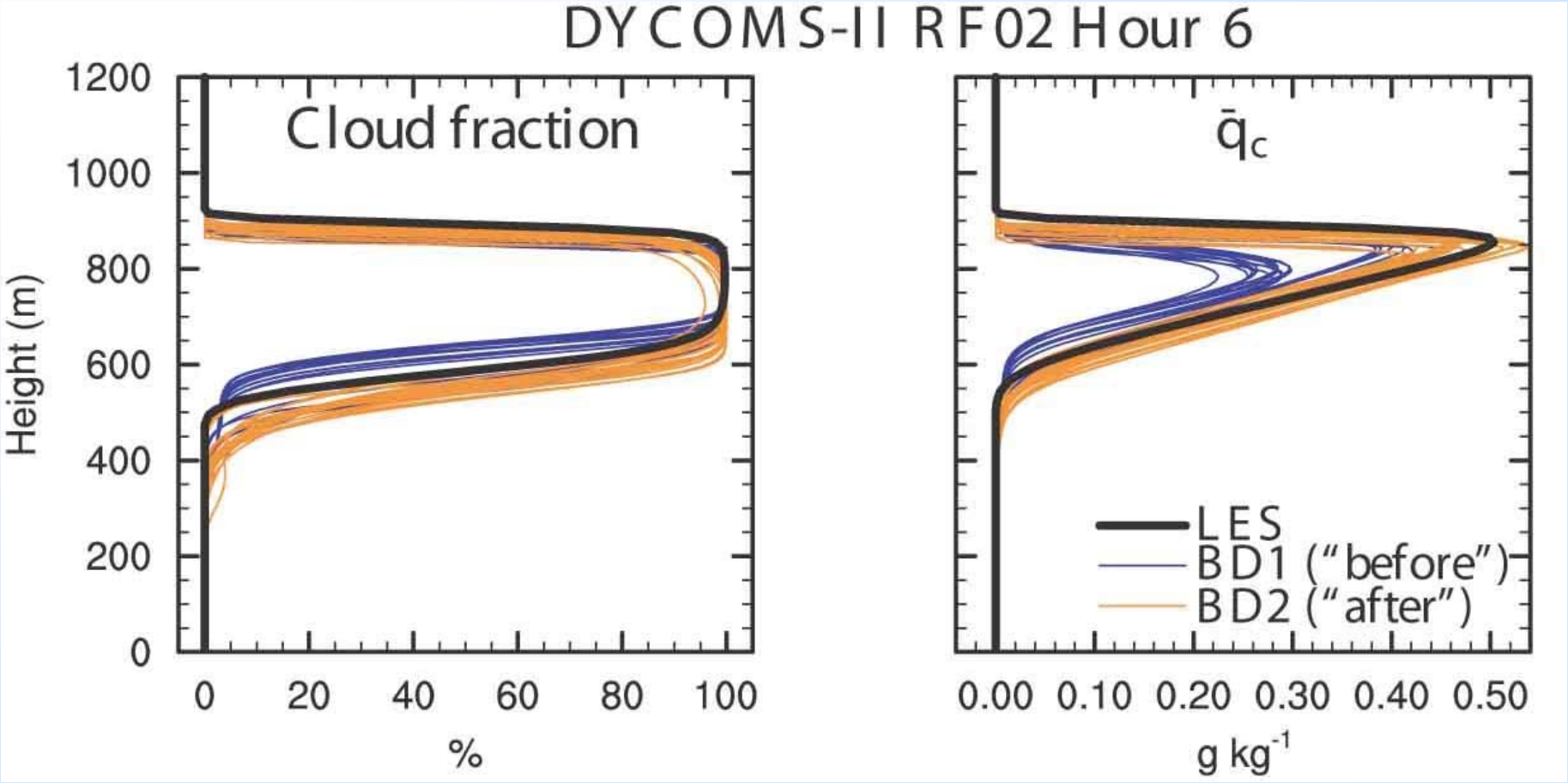
The ATEX Cu rising into Sc case doesn't change much



Cloud fraction

Liquid water mixing ratio

The RF02 marine Sc case improves



Cloud fraction

Liquid water mixing ratio

Thanks for your time!