Adaptive Monte Carlo sampling for cloud and microphysics calculations

by

Thomas Franz-Peter Roessler

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Mathematics

at

The University of Wisconsin–Milwaukee

May 2017

Abstract

Adaptive Monte Carlo sampling for cloud and microphysics calculations

by

Thomas Franz-Peter Roessler

The University of Wisconsin–Milwaukee, 2017
Under the Supervision of Professor Vincent E. Larson

An important problem in large-scale modeling of the atmosphere is the parametrization of clouds and microphysics on subgrid scales. The framework Cloud Layers Unified By Binormals (CLUBB) was developed to improve the parametrization of subgrid variability. Monte Carlo sampling is used to couple the different physical processes, which improves the grid average of subgrid tendencies. In this thesis we develop an adaptive Monte Carlo sampling algorithm that re-uses sample points of the previous time step by re-weighting them according to the change of the underlying distribution. This process is called "what-if sampling" and is an application of importance sampling. An example illustrates that what-if sampling converges slowly when the atmospheric conditions change too much. Therefore, the algorithm was extended by adaptive criteria. These prohibit re-weighting if the atmospheric conditions change too fast and allow the re-weighting method to converge to the right solution. We studied five test cases for different atmospheric conditions and found that the computation of the what-if weights is too expensive and suffers from bad importance sampling. The high-dimensional distribution of CLUBB that evolves in time makes re-weighting difficult. The simulation results of what-if sampling are similar to the standard Monte Carlo method or even worse considering the higher computational costs. Therefore, the algorithm was simplified such that old tendencies are re-used without any re-weighting. This approximation removes the overhead and reduces the extra noise. However, simple re-using does not improve the accuracy of the model for the same computation time for general appli-

cation. Only in the special case of very few sample points, this method can improve the performance without increasing the error significantly. The standard Monte Carlo sampler of CLUBB works very efficiently by applying well suited importance sampling. For normal simulations, using fewer sample points is better than applying any re-using algorithm to a larger number of sample points.

To my Family in Germany

Table of Contents

List of Figures

List of Tables

Acknowledgements

First, I would like to thank Dr. Vincent E. Larson for offering me a job as a research assistant in his science group. I really enjoy working in the field of atmospheric science. I also want to thank Professor Larson for advising me in this study and for keeping me motivated when results looked mittelpraechtig (middle-glorious).

Second, I want to thank Professor Hinow and Professor Stockbridge for being on my Master's thesis committee.

Then I want to give thanks for all the support of my family and friends from Germany. Especially to Michael and Vitalij for staying up late so often.

Chapter 1

Introduction

An important problem in large-scale modeling of the atmosphere is the parametrization of clouds and microphysics on subgrid scales. The spatial variability in clouds drives microphysical process rates and precipitation, which depletes clouds. Typical parametrizations account for the fact that a cloud may fill only part of a grid box, but they often assume that clouds are internally uniform or they make different assumptions for different cloud types. However, it is desirable if the parametrization accounts for within-cloud variability and if it is applied consistently for all simulated processes.

One approach that does consider subgrid variability and couples the different physical processes is Monte Carlo sampling. In the given framework Cloud Layers Unified By Binormals (CLUBB) this is achieved by the Subgrid Importance Latin Hypercube Sampler (SILHS). By sampling from different categories, SILHS provides a comprehensive coupling of clouds to all microphysical processes. Surprisingly accurate climatic averages can be found with relatively few sample points. For instance, the global simulations of Thayer-Calder et al. (2015) used only 10 sample points per grid box and time step. Nevertheless, this method is computationally expensive because it requires one call to the microphysics per sample point. With 4 sample points, the cost of the microphysics is increased by a factor of 4. The cost could be lessened by reducing the number of sample points, but then additional sampling noise would appear.

To reduce the cost, it may be possible to exploit the fact that at times localized atmospheric probability density functions (PDF), that are used for the parametrization of clouds, evolve relatively slowly. A quasi steady state might be expected to occur, for instance, if a shallow cloud layer is moistened by vertical turbulent transport, but this moistening is

balanced by loss due to precipitation. In such a situation it may be possible to re-use sample points, thereby saving calls to the microphysics. The PDF may be expected to change at least a little between time steps, and so the re-used microphysical tendencies cannot be applied as they are to the evolved PDF. Instead, the tendencies must be re-weighted.

A method for re-weighting the tendencies exists and is called "what-if" sampling. What-if sampling asks what would happen if old samples were applied to a new and different PDF. The new samples can be re-used if the change of the PDF is taken into account by re-weighting the old sampled tendencies. The method is most accurate when the new and old PDFs are similar. Here we apply what-if sampling to the special case of time evolving fields. In particular, we study 5 cases of precipitating shallow and deep clouds.

In chapter 2 we introduce the model and the mathematical background of Monte Carlo and importance sampling. Then the adaptive what-if sampling algorithm is developed and applied to the model. In chapter 3 we present a convergence test for the developed method. Thereafter, we present and evaluate the simulation results of what-if sampling for the five cases. We also present a couple of alternative approaches to reduce the computation time. In chapter 4 we conclude and discuss the results.

Chapter 2

Methods

2.1 The single column model CLUBB

The framework CLUBB simulates the evolution of clouds over time in a single vertical column of the atmosphere. While most climate models simulate the global weather in the three dimensional space, CLUBB focuses on one column of air that is modeled in more detail. Global models, for example, have a simple parametrization of the processes that take place on the subgrid scale, which is on the order of 100 km. A simple parametrization may include a cloud fraction for a grid box, but they do not consider local turbulent mixing or microphysics (Forbes, 2015). The framework CLUBB models the subgrid variability in more detail. It considers the different processes that take place on the subgrid scale and couples them in the computation of grid averages of the rate of change of atmospheric properties. For example, turbulent mixing and microphysical processes influence the total budget of time tendencies, which influences the complete simulation. The simulation quality can be improved with a better subgrid parametrization, because the processes can have a significant impact. Therefore, global models can integrate CLUBB to replace and improve their subgrid parametrization. One model that includes CLUBB for this purpose is the Community Atmosphere Model (Bogenschutz et al., 2013). The purpose of the adaptive sampling method is to reduce the computational costs of CLUBB such that it becomes attractive for practical application in more atmospheric models. When CLUBB is used as a submodule, some terms would be computed twice. Therefore, terms that are computed in CLUBB and the host model – like wind speed, air pressure, and large-scale moisture and temperature forcings – are given as an input parameter to CLUBB. Then the tendencies

of the subgrid processes are computed by CLUBB and sent back to the host model. The important terms of the subgrid scale that are computed with CLUBB are turbulent fluxes and time tendencies of the cloud microphysics. Turbulent fluxes are caused by eddies that cause mixing. Mathematically, turbulent fluxes are described as covariances between a velocity component and any quantity (American Meteorological Society, 2017). Most terms involve the vertical velocity and hydrometeors. Hydrometeors are species that are able to precipitate like rain, graupel, snow, or ice crystals. The term "cloud microphysics" refers to processes on the scale of the hydrometeor particles. Some examples are the evaporation of liquid water and the resulting cooling, the formation of rain drops due to condensation, or the collision and growth of particles.

Before the computation of turbulent fluxes and microphysics is described, we want to introduce CLUBB's subgrid parametrization. All types of clouds as well as clear conditions are modeled in the same way. The model uses a joint probability density function (PDF) of two multivariate normal-lognormal PDFs to model the distribution of clouds and the hydrometeors. Every vertical layer of the single column model has its own joint PDF that describes the atmospheric condition in the layer. As the model evolves in time, the distributions in the vertical layers change. The distributions are calculated with the assumed PDF method. For a more detailed description of the parametrization of CLUBB and the assumed PDF method we want to refer to Golaz et al. (2002). The equation for CLUBB's PDF is

$$P(x) = \sum_{i=1,2} \xi_i \Big[f_{p(i)} P_{norm}(\chi, \eta, w, N_{cn}, \mathbf{hm}) + (1 - f_{p(i)}) \delta(\mathbf{hm}) P_{norm}(\chi, \eta, w, N_{cn}) \Big]. \quad (2.1)$$

Where the sum represents two mixture components that are weighted according to ξ_i given $\xi_1 + \xi_2 = 1$ and $\xi_i \geq 0$. The model uses two mixture components to gain more freedom in modeling clouds. The function $f_{p(i)}$ gives the fraction of the precipitating area of the mixture components. The precipitating area is defined by the presence of at least one non-zero hydrometeor species. The function P_{norm} is a standard multivariate normal density, χ, η,

and w are given in normal space and N_{cn} and all hydrometeors – combined in the vector **hm** – are given in lognormal space. The first four variates are mandatory, they are χ moisture content, w vertical velocity, η temperature, and N_{cn} the number of cloud nuclei, which are small particles that allow water vapor to condensate more easily. These four variables are always present in the atmosphere, such that they are included in the precipitating and non-precipitating part of the PDF. The dimension of the multivariate normal PDF depends on the actual quantities that are modeled in the simulation. The number of different hydrometeors depends on the simulation. Usual configurations use eight hydrometeors which are mixing ratios and number concentrations for each of the four quantities rain, graupel, snow, and ice. Mixing ratios define how much mass in kg of the species is present in one kg of air and number concentrations give the number of particles in one kg of air. For example, the combination of both allows the modeling of different sized rain drops in different concentrations.

The computation of average microphysical tendencies and covariances for the turbulent fluxes takes place in the Subgrid Importance Latin Hypercube Sampler (SILHS) module. The module draws sample points from CLUBB's PDF and computes estimates for means and covariances. For each sample point the computationally expensive microphysics is evaluated. In this thesis we used the Morrison microphysics scheme (Morrison et al., 2005) for our calculations. The tendencies of the sample points are averaged which couples the different physical conditions in the simulated domain. The sample points are also used to compute variances and covariances. The process of drawing samples and computing averages and covariances is described in section 2.3.1.

2.2 Mathematical Background

2.2.1 Monte Carlo method

In this section we want to summarize relevant aspects of the Monte Carlo method and importance sampling, before we introduce the what-if sampling method. The Monte Carlo

method can be used to solve integral equations of the form,

$$\mu = \int f(x)P(x)\delta x, \tag{2.2}$$

where P denotes a PDF and f is any function. The PDF describes where the sample points are drawn and the function represents any quantity that we want to integrate, which does not need to be smooth or integrable. That is why Monte Carlo integration is used in CLUBB, the microphysics takes place in a complicated subroutine that cannot easily be integrated over the grid box. We will refer to PDF mass later on, which is simply the probability to draw a sample point in a specific region, that is given by

$$\mu = \int \mathbf{1}_{x\in C}P(x)\delta x, \tag{2.3}$$

where $\mathbf{1}$ is the indicator function and C is a subset of the domain of P, which is $\mathbb{R}^n$ for our case. Sample points are usually generated by drawing a uniform random number between zero and one and transforming it to the used PDF with help of the inverse of its cumulative distribution function (CDF). After sample points have been drawn, the estimate for the mean can be computed with

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(x_i), \quad x_i \sim P. \tag{2.4}$$

Where f and P are the same functions as above and n is the number of sample points drawn from the given PDF. Since all x_i are independent and identically distributed, the strong law of large numbers holds,

$$\lim_{n\to\infty} |\mu - \mu_n| \leq \epsilon \quad \forall\epsilon > 0 \quad a.s. \tag{2.5}$$

So we know that the estimate converges almost surely and any desired accuracy can be acquired by choosing a fitting number of sample points. Simple Monte Carlo estimation converges at a relative slow rate of $1/\sqrt{n}$. A measure for the goodness of the estimator is

given by its variance,

$$\mathbb{E}[(\mu_n - \mu)^2] = \frac{\sigma^2}{n}, \tag{2.6}$$

where σ^2 is the variance of $f(x)$ and n is the number of sample points. The smaller the variance the better is the estimator. The easiest way to reduce the variance is to increase the number of sample points, but there are other variance reduction technique that do not increase the computational effort. For more detailed information on convergence and the variance of the estimate we refer to Owen (2013).

2.2.2 Importance Sampling

Importance sampling is a variance reduction technique that can lead to significant improvements of the Monte Carlo estimator. This method is especially helpful in situations where a large fraction of the PDF mass is assigned to parts of the function that are close to zero and contribute little to the integral. The method tries to draw more sample points in the relevant parts of the function, by choosing a new source PDF for the sample points. The new PDF, called importance PDF, can be chosen almost arbitrarily and is supposed to sample preferably in regions where the integrand is large. When applying importance sampling, the integration equation changes as follows

$$\mu = \int f(x)P(x)\,dx = \int f(x)\frac{P(x)}{Q(x)}Q(x)\,dx. \tag{2.7}$$

Now we can draw sample points from the new PDF Q, but we have to re-weight the sample points according to the ratio of the original and new PDF. The equation for the importance sampling estimate is

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n}\frac{P(x_i)}{Q(x_i)}f(x_i), \quad x_i \sim Q. \tag{2.8}$$

The oversampling of the important parts of the function, resulting from the new PDF, is compensated by the re-weighting. A well-chosen importance PDF can improve the results

dramatically. The best results can be achieved when the new PDF is proportional to fP (Owen, 2003).

For importance sampling we have to compute the ratio P/Q, which causes two general problems. The first one is that we need to evaluate the density function at the sample points, which is not necessary for usual Monte Carlo sampling. The density function can be very costly and may not exist at all. The resulting values for the density function of Q have to be greater than zero wherever P is greater than zero. This condition is fulfilled in our cases, since P and Q are multivariate normal distributions, which are positive everywhere. The second problem is more substantial. The ratio of the PDFs can become huge if Q is much smaller than P, which can also increase the variance of the estimator instead of reducing it. This problem usually occurs in the tails of the distribution Q.

To illustrate the benefits of importance sampling, a test case is presented here. The PDF for this example was taken from simulation output and belongs to the rain water mixing ratio. We want to estimate the microphysical tendency for the rain mixing ratio for the conditions described by this distribution. The function for the microphysical tendency was obtained by exponential fitting of sample points from the simulation. The example is of illustrative nature and does not claim any physical correctness, since we want to give a one dimensional example and the microphysical tendency depends on all variates of CLUBB's PDF and not the rain mixing ratio alone. However, the scales of the mixing ratio and the tendency are realistic. The PDF for rain water mixing ratio was converted from the log-normal space to the normal space, such that $\mu = -11.5$ and $\sigma = 1.15$, which corresponds to a rain water mixing ratio of 1.01×10^{-5} kg/kg. In the test the average rain mixing ratio tendency is estimated with standard Monte Carlo integration and importance sampling. The importance sampling PDF is chosen more or less optimally by shifting the mean to the global maximum of the integrand, such that $\mu_{ip} = -10.7$. The estimate for the integral is computed 100,000 times with ten sample points each. Figure 2.1 illustrates the case

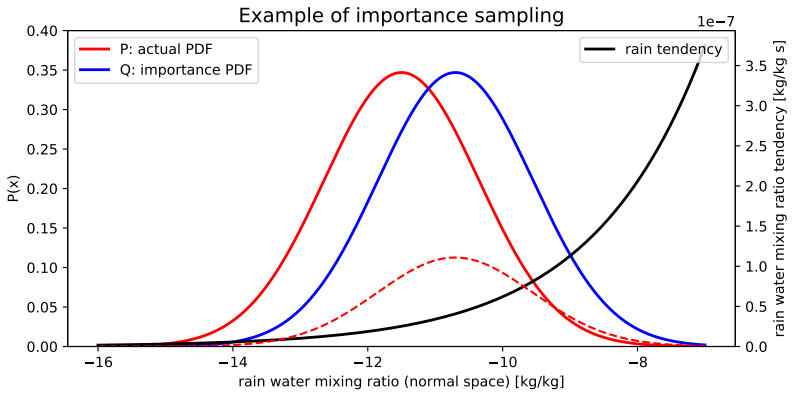

Figure 2.1: Example of importance sampling. The dashed line indicates the integrand scaled by a factor of 10.

setup and shows that the two PDFs do not differ much. Nevertheless, the small mean shift improves the estimate significantly. Averaged over all 1,000,000 samples, both methods give the right solution of about 3.197×10^{-8}. The interesting part is the variance of the estimator, which is equal to the averaged squared error of the estimates. The average error of the 100,000 estimates is much smaller for importance sampling. The average error drops from 6.14×10^{-17} to 3.26×10^{-21} when importance sampling is applied. The percentiles of the 100,000 simulations also illustrate the large improvement. The inner 80% of the estimates computed with standard Monte Carlo sampling are in the interval $[2.29\times10^{-8},\ 4.22\times10^{-8}]$, while importance sampling gives the interval $[3.19\times10^{-8},\ 3.20\times10^{-8}]$. In total over 99.5% of all importance sampling estimates were closer to the correct solution. The impact of importance sampling becomes even more clear by considering the number of sample points that is needed for standard Monte Carlo sampling to give similar good results. The standard deviation of the importance sampling estimator with 10 sample points is 3.26×10^{-21}. We now can use Equation (2.6) to compute the number of sample points that is needed for the standard Monte Carlo method to have the same variance. Given that the standard deviation of our function is 6.02×10^{-17} we get that 18,000 sample points are needed for the standard Monte Carlo method. This example shows that standard Monte Carlo sampling does not distribute the sample point optimally, which reduces the quality of the estimator.

2.2.3 What-if sampling

The purpose of what-if sampling is to re-use sample points and not to reduce the variance of the estimator. What-if sampling asks the question what would happen if sample points from one PDF were applied to another PDF. This is especially interesting in the scope of time evolving PDFs. Sample points from the old time step and an old PDF could be applied to the new time step with a different PDF to save the computation time needed to generate and evaluate the sample points. The theory of what-if sampling is based on importance sampling, which allows us to draw sample points from a different PDF than that given by the integrand.

At the first time step the integral is solved with standard Monte Carlo integration,

$$\int f(x)P_1(x)\,dx \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i), \quad x_i \sim P_1. \tag{2.9}$$

Here P_1 denotes the PDF at the first time step. Analogously, the true integral equation at the second time step is

$$\int f(x)P_2(x)\,dx. \tag{2.10}$$

Now we want to use the sample points of the previous time step that were drawn from P_1 instead of drawing new sample points from P_2. We can use the same argument as for importance sampling to get

$$\int f(x)\frac{P_2(x)}{P_1(x)}P_1(x)\,dx. \tag{2.11}$$

Then the estimate for the integral is

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(x_i)\frac{P_2(x)}{P_1(x)}, \quad x_i \sim P_1, \tag{2.12}$$

which uses the samples drawn from the old PDF P_1. This shows that what-if sampling is an application of importance sampling, where the main difference is that the PDFs are not chosen arbitrarily, but are fixed and given from the time steps of the simulation. We could

apply what-if sampling multiple times, such that P_1 is used as a reference for more than one time step. The number of calls to function $f(x)$ can be reduced significantly by re-weighting the old function values. However, we have to consider that importance sampling can worsen the estimator. In fact, in the scope of time evolving PDFs we apply something like anti-importance sampling, since the old distribution does most likely not describe the new conditions well. Instead of drawing more samples from the interesting region, we draw more sample points from the region that was important at the last time step. The less the PDFs change in time, the better results can be expected. The time step of CLUBB is in the order of 15 seconds to 15 minutes. Roughly speaking, weather does not change too much on the scale of minutes, so the distribution is assumed to change little enough to give good approximations.

2.3 What-if sampling algorithm in CLUBB-SILHS

2.3.1 Computation of the microphysical estimates

The grid averages for microphysical tendencies are computed in SILHS. The different physical schemes that are used for the parametrization of clouds, different types of precipitation or other phenomena are coupled with help of the Monte Carlo method. We draw multiple sample points that belong to different atmospheric conditions and evaluate the microphysics there, which allows us to consider the subscale variability for the mean calculation. The natural distribution of these sample points is given by CLUBB's PDF, but the actual calculation applies importance sampling to reduce the variance of the estimator. Importance sampling is applied by prescribing the probabilities to draw samples in specific categories, that are defined by meteorological conditions. The model can use two, four or eight categories. The first two categories prescribe probabilities for in cloud and out of cloud. The configuration with four categories splits the two existing categories into a precipitating and not precipitating category. The last configuration considers all four categories for both mixture components

separately, which gives us eight categories. We use the basic configuration with two categories in this thesis, because we want to use as few sample points as possible and a broad distribution of sample points could under-sample clouds, which are most important.

The probability to draw sample points from a specific category is

$$p_j = \int \mathbf{1}_{x \in C_j} P(x)\, dx, \tag{2.13}$$

where C_j is the category and P is CLUBB's PDF at the desired vertical layer. The categories are disjoint and span the entire range of the PDF, therefore the probabilities p_j add up to one. The goal of the importance sampling is to replace the probabilities p_j by a prescribed set of probabilities s_j, such that more important categories are sampled more often. The probabilities p_j depend on the meteorological conditions and change from layer to layer and from time step to time step. The prescribed probabilities s_j are constant for the simulation. SILHS applies importance sampling just for one reference layer explicitly, which is called *k_lh_start*. The layer with the most hydrometeors is chosen as the reference layer and defines the importance PDF for all layers.

The Monte Carlo estimate for the original integral equation of CLUBB is

$$\int f(x)P(x)\, dx \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i), \tag{2.14}$$

where n denotes the number of sample points of the estimate. We can split up the integration for the disjoint categories without changing the result, which gives us

$$\sum_{j=1}^{N_c} \int_{C_j} f(x)P(x)\, dx \approx \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{N_c} \mathbf{1}_{x_i \in C_j}\, f(x_i) \Big), \quad x_i \sim P, \tag{2.15}$$

where N_c denotes the number of categories and n is the number of sample points. The sample points x_i belong to exactly one category. Now we want to change the source PDF in such a way that the weights p_j become s_j. We can do this by multiplying and dividing our

PDF by $\mathbf{1}_{x_i \in C_j} \frac{s_j}{p_j}$ for each category, which gives us the new marginal distributions Q_j. The PDF Q is defined as the sum of the disjoint marginal distribution. If we multiply and divide Equation (2.15) by the marginal distributions we get

$$\sum_{j=1}^{N_c} \int_{C_j} f(x) \frac{p_j P(x)}{s_j P(x)} Q_j \, dx \approx \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{N_c} \mathbf{1}_{x_i \in C_j} \frac{p_j}{s_j} f(x_i) \Big), \quad x_i \sim Q. \tag{2.16}$$

Given that sample point x_i is drawn in category j, we can define the sample point weight,

$$\omega_i = \frac{p_j}{s_j}. \tag{2.17}$$

With what-if sampling we add an additional layer of importance sampling to Equation (2.16). We draw from distribution Q_1 that depends on P_1, of the first time step, but have given a distribution P_2 of the new time step. We multiply and divide the new integral equation by Q_1 and get

$$\begin{aligned} \sum_{j=1}^{N_c} \int_{C_j} f(x) P_2 \, dx &= \sum_{j=1}^{N_c} \int_{C_j} f(x) \frac{P_2(x)}{Q_1} Q_1(x) \, dx \\ &= \sum_{j=1}^{N_c} \int_{C_j} f(x) \frac{p_j}{s_j} \frac{P_2(x)}{P_1(x)} Q_1(x) \, dx. \end{aligned} \tag{2.18}$$

The new weight for a sample point x_i, given that it is was drawn in category j, is

$$\omega_i = \frac{p_j}{s_j} \frac{P_2(x_i)}{P_1(x_i)}. \tag{2.19}$$

The final estimator is the weighted average of the sample points.

2.3.2 Case study for what-if sampling

The application of what-if sampling is problematic if abrupt changes occur in the current time step – like the start or end of precipitation. In the following we want to give a one-dimensional example to illustrate how importance sampling will affect the estimated mean tendencies in the case of strong changes. We draw sample points from a PDF that fits to a case with moderate rain and use these samples to estimate the mean for PDFs which belong to light and heavy rain. The parameters for the PDFs in normal space are $\mu_l = -13$ and $\sigma_l = 1.1$ for light rain, $\mu_m = -11.5$ and $\sigma_m = 1.15$ for moderate rain, and $\mu_s = -11.5$ and $\sigma_a = 1.2$ for heavy rain. The matching mixing ratios for the means are 0.002 g/kg, 0.010 g/kg and 0.045 g/kg respectively.

Figure 2.2 shows the different PDFs and a rough application of the microphysical tendency obtaind by fitting an exponential function to simulation results. The figure also shows the product of the PDFs and the tendency function to show where samples contribute significantly to the integral. The exponential growth of the rain function causes the peak of the integrand for heavy rain to be far away of the PDF we draw from. This is a sign for slow convergence. For this test we used a sample size of 10 and repeated the calculation the integrals 100,000 times. The estimate for the mean of all three PDFs converges to the true solution, but the quality of the estimators, given by the variance, is not the same. The average errors and various percentiles of the estimates were computed and are presented in Table 2.1.

The average estimate for the mean is very close to the real solution for the cases with light and moderate rain. The variance of the estimators is small which shows that the results are relatively robust and do not suffer from high noise. The case with heavy rain also gives an estimate in the order of the real solution, but the variance is many times larger. The higher variance for the test cases that use re-weighting comes from the larger variability between the sample points. Figure 2.3 shows the weighted sample point values for the sample points on a logarithmic scale.

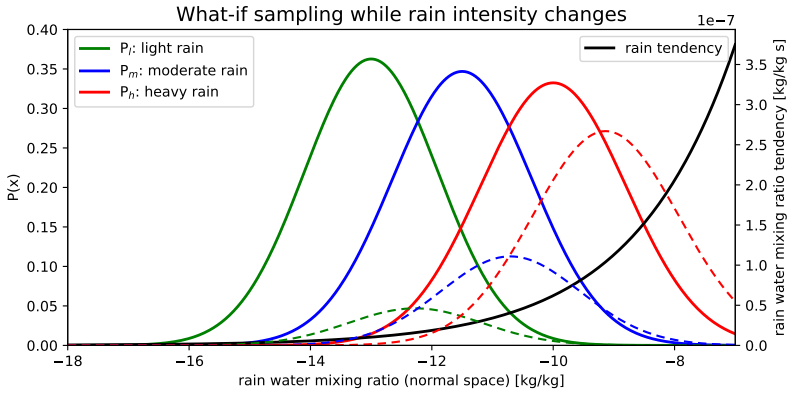

Figure 2.2: Illustration of the PDFs for light, moderate and heavy rain. The PDF of moderate rain is used to draw samples to estimate the rain tendency. Dashed lines show the product of the rain function and the PDFs, scaled by a factor of 10.

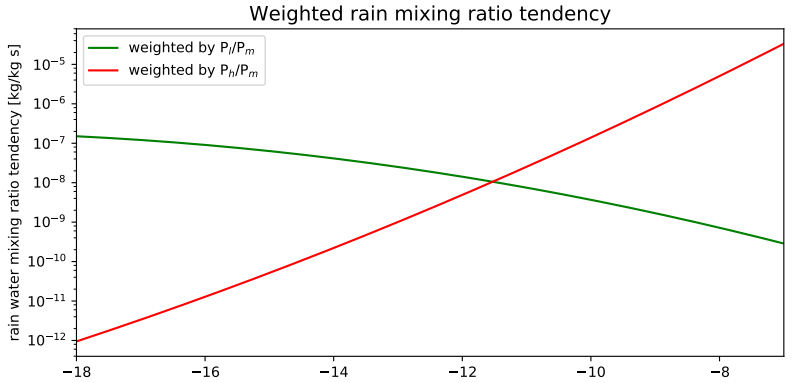

Figure 2.3: Weighted tendencies for the sample points drawn from the PDF belonging to moderate rain.

	light rain	*moderate rain*	*heavy rain*
real solution	1.27e-08	3.20e-08	8.03e-08
mean estimate	1.27e-08	3.20e-08	8.10e-08
variance	8.56e-18	6.26e-17	6.04e-14
$Q_{.10}/\mu$	0.7	0.7	0.2
$Q_{.25}/\mu$	0.8	0.8	0.3
$Q_{.50}/\mu$	1.0	1.0	0.5
$Q_{.75}/\mu$	1.1	1.1	1.0
$Q_{.90}/\mu$	1.3	1.3	1.9
$Q_{.95}/\mu$	1.4	1.4	3.0
$Q_{.99}/\mu$	1.6	1.7	7.9

Table 2.1: Mean solution and variance of the estimates of the rain tendency for light, moderate, and heavy rain. The quantiles were divided by the real solution to illustrate the deviation.

For the case with light rain, the sample point values drop relatively slowly from 10^{-7} to 10^{-9}. This means that the sample points contribute more or less equally to the integral. The sample points with near zero values for $x > -10$ do not hurt the estimator, because only few samples are drawn there. So the area with significant weights roughly corresponds to the area where the integrand is large, that is why the estimate for the light rain case is relatively good. The quantiles show that the median for the estimate in the order of the real solution and the deviations of the other percentiles are the same or even better than those for the standard Monte Carlo estimate. So, for the what-if simulation in CLUBB we expect relatively good estimates for the tendencies when rain decreases.

For the test case with heavy rain the weighted sample values show a complete different behavior. The weights for the sample points grow faster, but all sample point with $x < -11.5$ have a near zero contribution to the integral. So about 50% of all sample points are located in an unnecessary region for the case with heavy rain. As the contribution of a sample point increases, the probability of that sample point decreases. So samples with large function values are sampled relatively rarely. However, this will happen from time to time and give a huge value for a sample point, which will then cause an overestimation of the mean. Nevertheless, the estimator will convergence, if enough sample points are considered. The problem is that the sample size of CLUBB is relatively small, which makes the estimates very noisy. The 75% quartile of the estimates for heavy rain is of the same order as the real solution. This means that 75% of all estimates underestimate the integral. On the other hand, 10% of the estimates are at least 2 times larger and 1% is even 8-400 times larger than the real solution. From this it follows that cases with increasing rain will have relatively noisy rain tendencies. It can happen that the rain tendency is underestimated for many time steps, which would delay the rain. On the other hand it can also happen that the rain starts instantly, because the tendency is strongly overestimated.

In conclusion we can say that the re-weighted sample points converge to the right solution, but many sample points are needed when the PDFs are too different, because we have to

counteract the increasing variance due to anti-importance sampling. That is the reason why what-if sampling was implemented adaptively. We have to make sure that the conditions are well suited for re-weighting.

2.3.3 The adaptive criteria

The previous example has shown that a straight forward implementation of what-if sampling, that re-uses and re-weights for a fixed number of time steps, can suffer from strong noise. The rate at which the PDFs changes varies through the simulations. The distributions can stay constant for some time and then change rapidly. When the number of reuses is adaptive, constant parts of the simulation can be sped up while difficult conditions do not suffer from under-sampling of the microphysics. A difficulty for the adaptive what-if sampling method in CLUBB is that the sample points for all layers have to be generated at once. Considering that more than 100 vertical layers are given, it is very likely that strong changes occurred at least at one layer. The adaptive method has to remove dangerous spikes but also redraw as infrequently as possible. Therefore, the adaptive criteria work on two stages. The first stage identifies layers that indicate bad re-weighting and marks them as invalid. In the second stage, a decision for re-using or drawing new sample points is make. When sample points are re-used, the valid layers get updated according to the computed weights, but the invalid layers cannot be re-weighted. We implemented different strategies for updating the invalid layers, that are based on different assumptions. The first strategy assumes that the changes in the microphysical tendencies from one time step to the next are relatively small. Therefore, the weights are set to one for invalid layers, which means that old tendencies are used without any re-weighting. The second approach uses the layer where importance sampling is applied as a reference and re-weights all invalid layers with the weights of this layer. The third and last strategy is based on the assumption that the distributions of two layers that lie on top of each other are similar, so invalid layers use the weights of the nearest valid layer. The problem with all strategies two and three is that new

random samples are used in every vertical layer. SILHS correlates samples in the vertical to some extend, but tests have shown that the weights of different layers are too different. Therefore, we use the first strategy and use the old tendencies for layers with invalid weights.

Now we want to present the three criteria that are used to mark the invalid layers, that would apply bad re-weighting. These criteria are based on the means and standard deviations of the original and current PDF as well as the what-if weights of the sample points.

The first criterion makes sure that the PDFs of the reference and current time step are valid and that their densities exist. Not all variates of the multivariate normal distribution are present at all times. For example, all variates that model precipitation are not present above the cloud or when the sky is clear. These zero-variates of the PDF are modeled as a delta function at zero, which has zero standard deviation. From this it follows that the density of the full PDF does not exist. However, since all variates of the delta function are zero, we can use the subspace of all variates with non-zero standard deviation. This criterion makes sure that all mandatory variates have a valid standard deviation. In addition the criterion checks that the reference and the PDF of the new time step have the same valid variates. If the new distribution has more or less delta functions, it indicates that the meteorological conditions changed significantly and we draw new sample points to prevent over- or under-sampling of specific variates. The decision if a variate will be used depends on two conditions. First, the standard deviation must be greater than zero, which is a necessary condition for the computation. Second, the mean of the variate must be larger than a physically oriented tolerance that is chosen for all variates individually. The second condition was added for performance tuning. The simulations is not influenced substantially by quantities that have relatively small values, so they are left out for the what-if sampling. This is especially relevant for the performance of the second criterion.

The second criterion analyzes how strongly the current PDF differs from the reference time step. We check if the difference of the means of the two distributions are smaller than a

maximum change. The maximum change is based on the standard deviation of the reference time step. We allow a change of 1.0 times the standard deviations of the quantities.

The third criterion is applied after the new weights have been calculated. Here we analyze if the new estimate for the mean will be dominated by a few heavy weighted sample points. The used metric is the effective sample size

$$n_e = \frac{\left(\sum_{i=1}^{n} \omega_i\right)^2}{\sum_{i=1}^{n} \omega_i^2}, \tag{2.20}$$

where ω_i are the weights of our sample points. The effective sample size a number between one and the number of sample points. The larger it is, the more evenly weighted are the sample points. When the weights show that only a few samples contribute to the mean effectively, the possibility of falsifying the mean is large. One the other hand, a large effective sample size only means that sample points are weighted similarly, but not that we sample in all interesting regions. So, a large effective sample size is no guarantee for good importance sampling. We defined the minimum effective sample size to be the square root of n, which is the total number of sample points.

The final decision if what-if sampling is applied depends on two criteria. A first trivial criterion makes sure that the sample points are not older than a maximum number of time steps. We want to that new sample points are drawn after a defined number of time steps, to ensure that unfortunate sample points are not used for too long. Actually, this criterion is evaluated at the beginning, before what-if sampling does any work. The second criterion is evaluated when the valid layers for re-weighting are known and the new weights have already been calculated. In the final check, the valid layers above and below our reference layer are counted. If at least half of them allow re-weighting, sample points are re-used. The number of layers that are taken into account above and below can be configured by the user. It basically depends on the test case and the depth of the clouds. When the analyzed layer

depth is set to zero, only the reference layer defines whether we should re-weight or re-draw. This configuration turned out to give good speedups and relatively good solutions and is used as a default.

2.4 Test case evaluation

The evaluation covers five case studies that are related to different cloud or weather experiments and are used for model validation purposes. The first two case studies are ARM 97 (Ackerman et al., 2004) and TWP-ICE (May et al., 2008). They produce relatively large amounts of precipitation and are expected to be more challenging, because the multivariate distribution has more variates. The other three test cases are DYCOMS 2 (Stevens et al., 2003), MPACE-B (Verlinde et al., 2007) and RICO (Rauber et al., 2007), which produce no or not much precipitation. The evaluation of the what-if sampling algorithm will compare the solutions of the new what-if sampling algorithm to the previous model code. The simulation results can vary from run to run, therefore ensembles of simulations are computed for the what-if sampling and standard SILHS implementation. The simulations using SILHS are called control simulation and the simulations using what-if sampling are called test or what-if simulations. The ensembles of control and test simulations are compared to a benchmark simulation, that was computed with 1024 sample points and is assumed to be the correct answer. One part of the comparison is done visually, by showing the minimal and maximal solution of the ensemble as well as the average of the solutions. The other part of the evaluation is done with help of time integrated errors between the control or test simulation and the benchmark simulation. We use the fields liquid cloud water path (LWP) and rain water path (RWP) as an indicator for overall simulation quality. The LWP is the amount of liquid water that is inside of clouds integrated over the vertical column measured in kg/m^2. Analogously, the RWP is the vertical integral of rain water also measured in kg/m^2. The LWP and RWP were used as indications, because all processes influence the vertical inte-

grals. The error of the control and what-if simulations to the benchmark simulations were computed with

$$Err(x, \bar{X}) = \frac{1}{t_{max} e_{max}} \sum_{t=1}^{t_{max}} \sum_{e=1}^{e_{max}} |x_e(t) - \bar{X}(t)| \tag{2.21}$$

where t_{max} is the number of time steps and e_{max} is the number of ensemble members . The function $\bar{X}$ returns the average solution of the benchmark simulations and x_e gives the result of an individual ensemble member.

Chapter 3

Results

3.1 Convergence tests

The adaptive what-if sampling algorithm was tested by a convergence test to validate that the re-weighted tendencies converge to the right solution. The algorithm works correctly if the re-weighted tendencies of the what-if step converge to the tendencies that would have been obtained with new sample points. For the convergence test the model was configured to compute the what-if tendencies and tendencies of new sample points at every time step, such that the results can be compared. The what-if tendencies are always computed based on the PDF of the previous time step, which means that sample points are only reused once in this test. However, we get a new what-if estimate at every time step, because the usual SILHS method is called also, which generates a new reference for the re-weighting. The tendencies of the new sample points feed back into the simulation and drive it forward, while the what-if sampling results are just written to disk.

We chose the RICO and TWP-ICE test cases for the convergence test to cover different levels of cloudiness and precipitation. All adaptive criteria are turned on as described in section 2.3.3, such that bad re-weighting is detected and prevented by using the old tendencies. However, the convergence test does not consider re-used tendencies that were not re-weighted. The what-if estimates for the average microphysical tendencies are based on the re-weighted tendencies alone, since we want to validate the re-weighting algorithm. The simulations used 16 to 2048 sample points and a time step of 5 minutes. The RICO test cases allowed re-using in 99% of all time steps and included 99% of all layers with relevant amounts of hydrometeors in the re-weighting process. The TWP-ICE case is more restric-

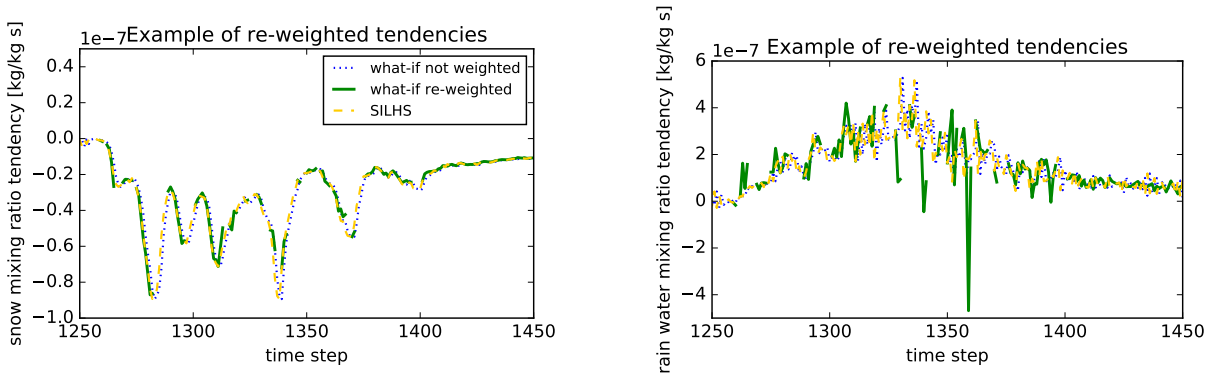

Figure 3.1: Example of re-weighted tendencies for rain and show mixing ratios at a single layer at 3.6 km altitude. The shown TWP-ICE simulation used 2048 sample points.

tive, the fraction of re-uses increased from 33% for 16 samples to 61% for 2048 samples and the fraction of valid layers increased from 65% to 85%. Considering that TWP-ICE includes physical processes for grauple, snow, and ice, which are all zero for RICO, the rates of re-using and re-weighting are good for both cases.

We want to start with giving an example of the re-weighted tendencies at a single layer to illustrate how the estimated tendencies mimic the curve shape of the solution of SILHS. Figure 3.1 shows a time series of the tendencies for snow and rain water mixing ratio at 3.6 km altitude of the TWP-ICE case. The figures show the estimate of SILHS (dashed orange), re-weighted tendencies (green line) and the tendencies of the previous time step (dotted blue), which are used if the re-weighting is prohibited. The physical conditions for the time series cause snow to melt to rain at the given layer, therefore we have a negative snow mixing ratio tendency and a positive rain water mixing ratio tendency. The curve shape for the snow mixing ratio tendency is relatively smooth, which indicates that this variate of the distribution changes little between time steps. We can see that the re-weighted tendencies are usually closer to the solution of SILHS than the old tendencies. Thus, the re-weighting for the snow mixing ratio works well, no outliers are present and the re-weighted tendencies match the expected values. The rain water mixing ratio tendency computed by SILHS is much noisier and does not have a smooth shape. The reference solution of SILHS is not

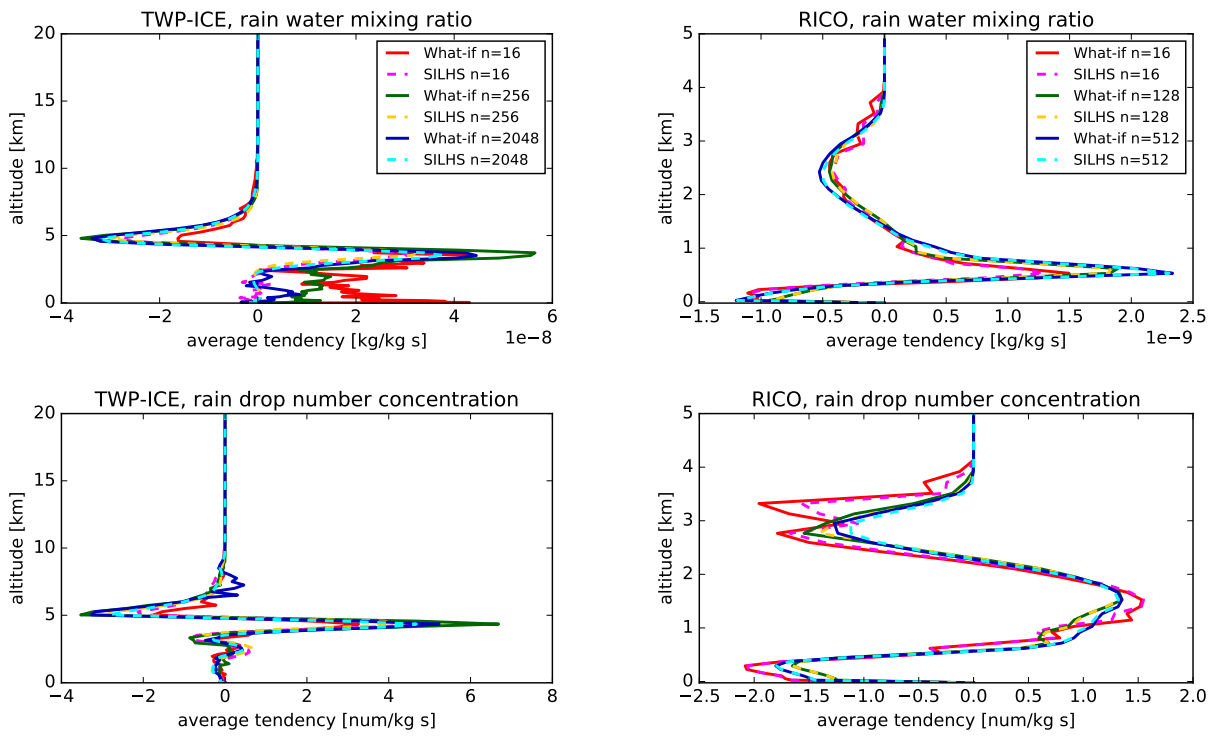

Figure 3.2: Average tendencies for rain mixing ratio and rain number concentration.

converged perfectly, since the tendency oscillates between time steps. The abrupt changes of the reference cause the what-if tendencies to be noisy and defective. Considering that large outliers appeared, the un-weighted tendencies give a better estimate overall. The different quality of the re-weighted tendencies for rain and snow is related to bad importance sampling. Since the PDF for rain is noisier and changes faster, the reference and new PDF are more different and anti-importance sampling is applied. We can also apply the findings of the case study for the change of rain intensity to this convergence test. Snowfall is decreasing and allows good re-weighting, while the rain increases and makes re-weighting difficult.

Noisy estimate for the time tendencies are not that problematic, as long as they equal out over time and do not create a bias. The turbulent fluxes and microphysical processes usually make up a comparatively small part of the total budget of time tendencies, such that oscillations will not harm the simulation stability and quality instantly. Therefore, layer average tendencies are used for this convergence test. Figure 3.2 shows the results for the rain mix-

ing ratio and rain number concentration. The average of the RICO case converges relatively fast, only the simulation with 16 sample points has some divergence from the right solution. The case TWP-ICE converges much slower, 2048 sample points and more are needed to get the same results for the re-used and new tendencies. The overall curve shape is already visible for 16 sample points, but the magnitude of the average is bad at the beginning and improves with an increasing number of sample points. For TWP-ICE the lower part of the rain mixing ratio tendency converges slow and is relatively noisy. This indicates that rain can be a weakness of the re-weighting as described earlier. For completeness, the average tendencies for the other hydrometeors of the TWP-ICE case are given in Appendix A. The estimate for all hydrometeors improve and approach to the solution of SILHS, however, at different speeds.

In conclusion, the average of the what-if re-weighted tendencies converges to the solution of SILHS. Nevertheless, without the filter criteria of the adaptive algorithm the results would be falsified by outliers such that the results are completely ruined. The anti-importance sampling causes the estimates to be noisy when the conditions are too different. It is possible that the tendencies of the previous time step are better estimates than the re-weighted tendencies when the conditions are difficult.

3.2 What-if sampling

In this section we want to present and evaluate what-if sampling simulation results for the five cases. We start with a set of simulations that used 32 sample points for the what-if and control simulations. All simulations were computed 20 times with different start seeds for the random number generator. The following figures will show the ensemble average and the min and max values, which are given by a shaded area. We consider the min and max range rather than percentiles, because filtering the data would remove too many errors. Large errors usually last only for a few time steps and the errors of the ensemble members do not overlap most of the time. The min and max range shows us how often large errors occur and if the simulation can recover.

Figure 3.3 shows the LWP for the benchmark, control, and what-if simulations. The corresponding figures for RWP are given in Appendix B. The results for the ARM 97 case are good for the control and what-if simulation. Especially the first half of the simulation, which has comparatively little cloud water, is solved well. The means match the benchmark and the spread is small. The second half shows a larger spread, but the means are still close to the right solution. The mean of the what-if simulations oscillates with a larger amplitude and we can find a couple of large errors. However, the control simulation has errors of about the same magnitude, they are just a little less frequent. Therefore, based on the plots the simulation quality does not decrease much for ARM 97. The cases DYCOMS 2 and MPACE-B are relatively short and easy cases that can be solved easily with 32 sample points. Therefore all lines fall more or less onto each other. The RICO case is also relatively easy, the results may look noisy, but the absolute error is small. The TWP-ICE case has the highest amounts of hydrometeors. The averages for the control and what-if simulations are very similar and close to the right solution. The spread of the control and what-if simulations is very similar. Large errors occur at about the same time steps and are of the same order.

So far we could not detect relevant differences between the simulations, ensemble means and the spread of the simulations are too similar. For a more objective analysis we want to

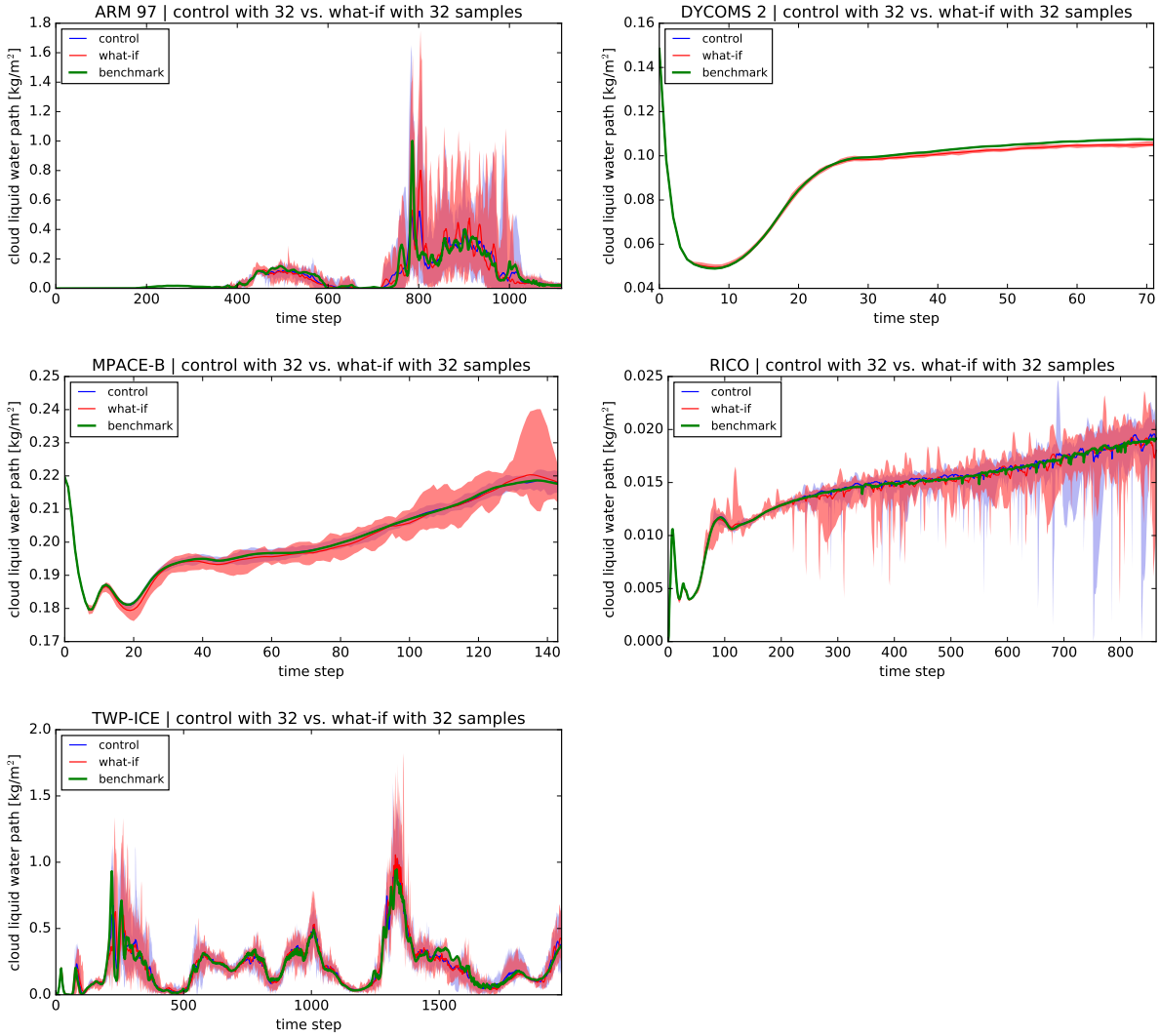

Figure 3.3: Results for LWP for what-if sampling and 32 sample points.

		costs		*avg. error LWP*		*avg. error RWP*	
case	*# samples*	*physics*	*total*	*control*	*what-if*	*control*	*what-if*
ARM 97	32	57 %	68 %	4.15e-02	4.80e-02	1.90e-02	2.62e-02
DYCOMS 2	32	36 %	42 %	1.35e-04	1.25e-03	3.26e-05	1.48e-04
MPACE-B	32	45 %	55 %	6.64e-04	1.61e-03	8.97e-05	2.29e-04
RICO	32	34 %	46 %	4.75e-04	6.00e-04	1.01e-03	1.18e-03
TWP-ICE	32	85 %	102 %	4.96e-02	5.53e-02	4.40e-02	5.97e-02

Table 3.1: Computation time and average errors of what-if simulations in comparison to control simulations with 32 sample points.

consider the average ensemble error, given by Equation (2.21). Table 3.1 shows the average errors for LWP and RWP. Obviously, the errors for the what-if simulations are larger, since less samples are computed. The error in LWP increased by 14% for ARM 97 and 10% for TWP-ICE, while the error in RWP increased by 36% and 38% respectively.

The relative growth for the three other cases is relatively large, but the absolute errors are still negotiable. We want now want to put the errors in relation with the different computation times for the what-if and control simulations. The difference in computation time is also given in Table 3.1. The reduction of the microphysics costs varies strongly from case to case. RICO and DYCOMS 2 require about 35% of the computation time that is needed by the control simulation, while ARM 97 and MPACE-B have moderate speedups and TWP-ICE is only slightly faster in terms of microphysics. A physics cost of 33% means that all sample points are used for three time steps and re-weighted twice. Physics cost of 50%, 67%, and 80% mean that sample points are re-weighted for 1, 0.5, and 0.25 times on average, respectively. We compare the computation time instead of the actual number of calls to the microphysics, because we want to consider the overhead. The re-weighting is relatively expensive and reduces the speedup notably. This is especially true for the TWP-ICE case, which is slower than the control due to the re-weighting. What-if sampling seems to be applicable for the three easy cases, since the computation time can be cut in half without getting significant errors. The ARM 97 case has a moderate speedup and a small additional error in LWP, but the error in RWP grew relatively strong.

		costs		*avg. error LWP*		*avg. error RWP*	
case	*# samples*	*physics*	*total*	*control*	*what-if*	*control*	*what-if*
ARM 97	12	87 %	103 %	4.75e-02	5.45e-02	2.62e-02	3.11e-02
DYCOMS 2	20	88 %	103 %	2.77e-04	1.31e-03	7.00e-05	1.70e-04
MPACE-B	13	75 %	93 %	1.52e-03	2.34e-03	2.05e-04	3.13e-04
RICO	18	74 %	101 %	7.66e-04	7.65e-04	1.31e-03	1.34e-03
TWP-ICE	9	92 %	111 %	7.03e-02	7.47e-02	7.84e-02	8.54e-02

Table 3.2: Computation time and average errors of what-if simulations in comparison to control simulations with 8 sample points.

The previous experiment did not consider that a reduction of the sample points with standard SILHS might lead to equally good results with the same costs. Therefore we present a second set of simulations that required the same computation time. We set the number of sample points to eight for the control simulations and adapted the number of sample points for the what-if simulations to keep the computation time constant. Table 3.2 shows the number of sample points, the difference in computation, and the average errors for the simulations. Instead of 8 the what-if simulations used 9-20 sample points. However, even if more sample points are used the effective number of calls to the microphysics is lower for the what-if simulations. The average and min and max results are shown in Figure 3.4. Again a evaluation is rather difficult. The mean estimates did not worsen much compared to the ensemble with 32 sample points. The spread of the solutions increased more notably for the what-if simulations, but there are still regions in which the control simulation is worse. However, the average errors for the control simulation are smaller in general, which shows that what-if sampling cannot improve the results efficiently.

The second experiment makes clear that what-if sampling cannot reduce computation time by maintaining the quality of the results. It is more efficient to reduce the number of sample points than to use more sample points and re-use them. The additional noise of the re-weighting, caused by the anti-importance sampling, is more problematic than the

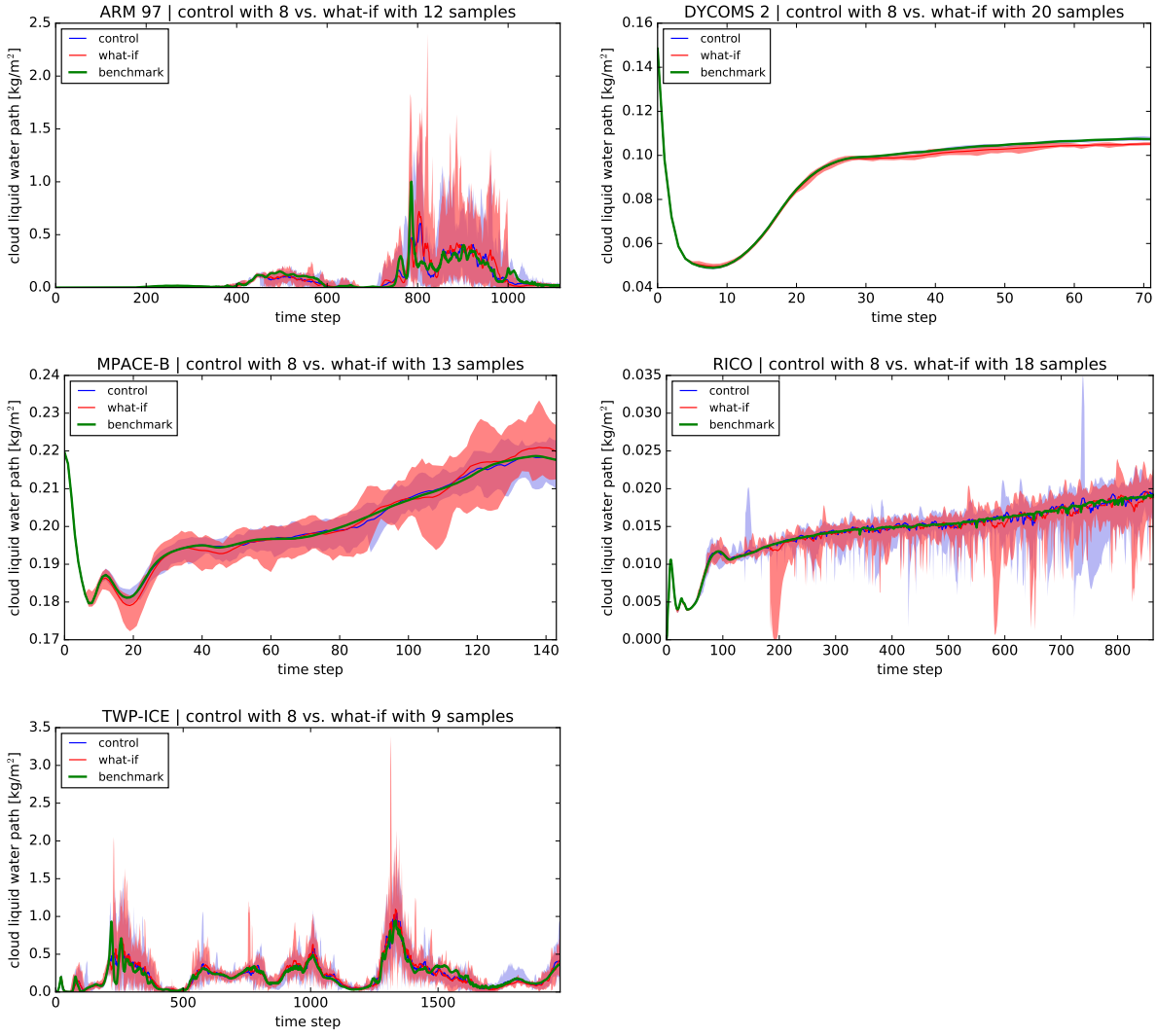

Figure 3.4: Results for LWP for what-if sampling and equal computation time.

SILHS estimator, which has a larger variance with fewer sample points. Another point is that the converge rate of Monte Carlo is very slow, such that a few more sample points will not improve the estimate much. On the other hand, fewer sample points do not degrade the estimator significantly. However, this gives one possible area of application for the what-if sampling. In the limit of very few sample points what-if sampling could be efficient. For example re-weighting with 3 samples could be more efficient than standard SILHS with two samples, because the variance of the estimator differs more strongly. We will consider this situation in Section 3.4. Before that, we will try to improve the potential of what-if sampling by reducing the overhead. It turned out that it is better to invest the computation time in additional microphysics evaluations. Therefore, we will simplify the what-if sampling algorithm in Section 3.3 and see if better results can be obtained.

3.3 Re-using without weighting

The simulations of what-if sampling showed that the method is not computationally efficient. Therefore, we simplified the algorithm by omitting the re-weighting, which eliminates the overhead. The disadvantage is that old tendencies are re-used as they are without any adjustment to the new conditions. Re-using old tendencies is reasonable if the meteorological conditions did not change too much, which is also a requirement for what-if sampling anyway. The advantage is that we will not get extra noise caused by the bad importance sampling. However, we cannot expect converge of the tendencies at the new time step. Applying the same tendencies for two time steps is not the same as applying them once with a larger time step, because other parts of the model are evaluated twice. For comparability, we want to compute the same types of experiments like presented in Section 3.2. The re-using algorithm is also adaptive. It uses the criterion for the maximum number of re-uses and checks the mean shift of the reference layer. The overhead costs for this algorithm are technically zero and additional sampling noise due to anti-importance sampling will not occur.

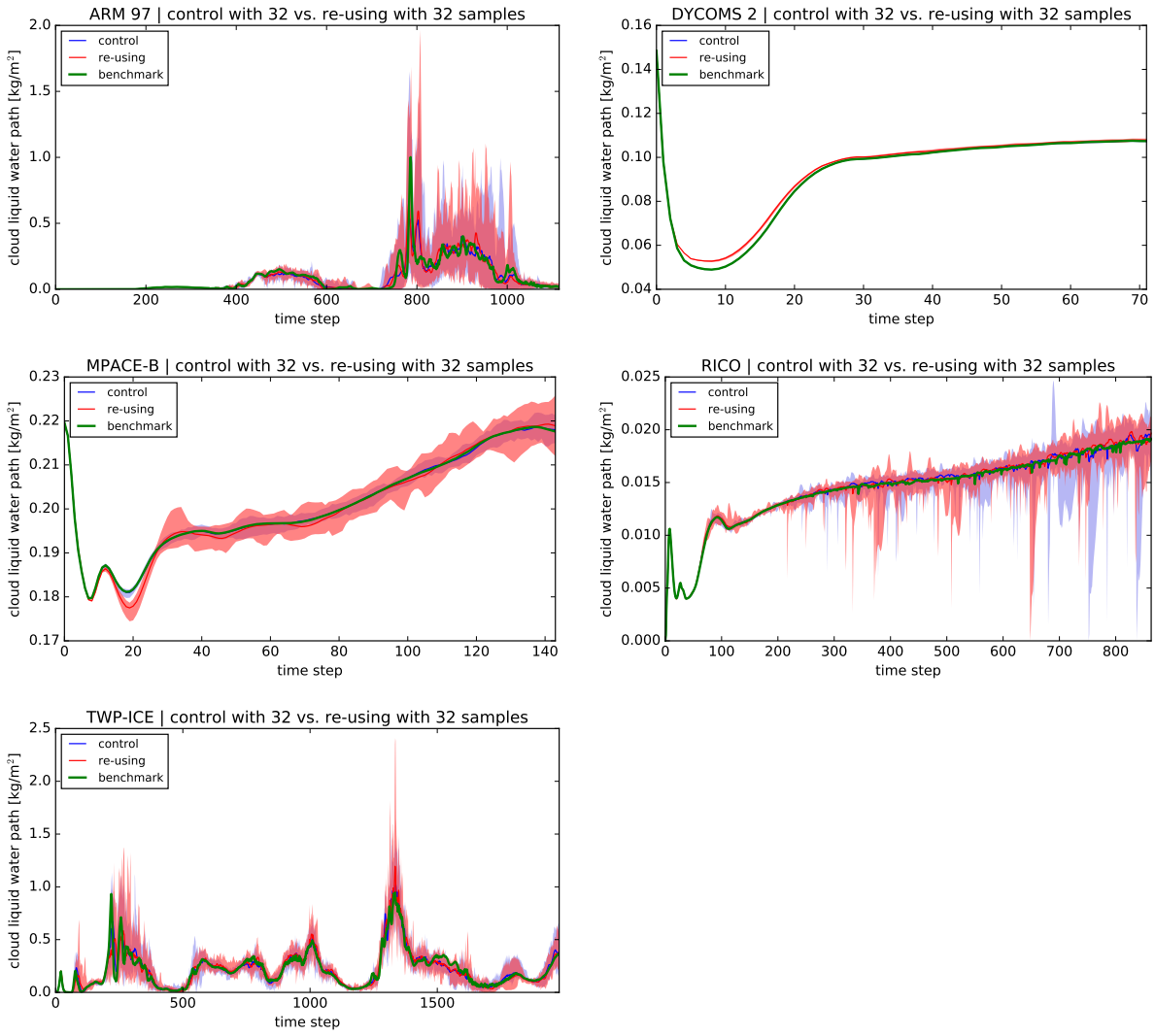

Figure 3.5: Results for LWP for the re-using algorithm and 32 sample points.

We want to start with the simulations that used 32 sample points. Figure 3.5 shows the ensemble average and the min and max solution for LWP. The spread of the solutions is smaller than for what-if sampling. The ensemble mean of the ARM 97 case even improved notably. The results for the easy test cases are similar to the previous results. The DYCOMS 2 is a little worse in the beginning, but no considerable differences are visible. The solutions for TWP-ICE also do not differ much, the largest spike was removed by simple re-using, but the overall performance is more or less the same.

		costs		*avg. error LWP*		*avg. error RWP*	
case	*# samples*	*physics*	*total*	*control*	*re-use*	*control*	*re-use*
ARM 97	32	74 %	74 %	4.15e-02	4.24e-02	1.90e-02	2.19e-02
DYCOMS 2	32	37 %	37 %	1.35e-04	1.36e-03	3.26e-05	2.13e-04
MPACE-B	32	44 %	44 %	6.64e-04	1.63e-03	8.97e-05	2.58e-04
RICO	32	57 %	58 %	4.75e-04	5.06e-04	1.01e-03	1.06e-03
TWP-ICE	32	82 %	82 %	4.96e-02	5.68e-02	4.40e-02	6.40e-02

Table 3.3: Computation time and average errors of the simple re-using algorithm in comparison to control simulations with 32 sample points.

Table 3.3 gives the computation time and the average errors. For DYCOMS 2, MPACE-B, and TWP-ICE the computation time is about equal to the computation time of what-if sampling without the overhead. The speedup for ARM 97 and RICO went down a little. With simple re-using, the average errors went down for ARM 97 and RICO and increased for DYCOMS 2, MPACE-B, and TWP-ICE. So the cases that are more accurate now are a little more expensive than before. The simple re-using method looks most promising for ARM 97, which was sped up by 25% by increasing the error to the control simulation by only 2% in LWP and 15% in RWP. The corresponding errors for TWP-ICE increased by 15% and 45%. We see that the rain water path is more difficult to compute when sample points are re-used, regardless of if they are weighted or not.

Now we want to compare control and re-use simulations with the same costs. The control simulation used 8 sample points and the number of sample points was increased to 10-20 for the re-use simulations. In general, the number of sample points for the re-using algorithm is different from the number of samples for what-if sampling. One reason is that the overhead time could be ignored which allows more samples, but the adaptive criteria also influence the simulations differently. Table 3.4 gives the information about sample points, computation time and average errors. The cases ARM 97 and RICO have smaller errors for the re-using than for the control simulation. However, the computation time is also a little higher for

		costs		*avg. error LWP*		*avg. error RWP*	
case	*# samples*	*physics*	*total*	*control*	*re-use*	*control*	*re-use*
ARM 97	11	105 %	105 %	4.75e-02	4.63e-02	2.62e-02	2.69e-02
DYCOMS 2	20	94 %	94 %	2.77e-04	1.43e-03	7.00e-05	2.09e-04
MPACE-B	17	96 %	96 %	1.52e-03	2.05e-03	2.05e-04	3.09e-04
RICO	14	102 %	102 %	7.66e-04	6.74e-04	1.31e-03	1.27e-03
TWP-ICE	10	97 %	97 %	7.03e-02	7.30e-02	7.84e-02	8.73e-02

Table 3.4: Computation time and average errors of the simple re-using algorithm in comparison to control simulations with 8 sample points.

this two cases. The errors for the other cases did not increase much for the re-using. For completeness, the spread of the solutions shown in Figure 3.6. We see that the spread for the simulations of the control and re-use simulations is really similar.

If we compare the results for re-using and re-weighting, we see that re-using tends to give better results in general. The results for ARM 97 and RICO are better for the re-using and the other cases are of more or less equal quality. However, the effective number of calls to the microphysics is significantly higher for the re-using. This shows that the overhead of the re-weighting makes the method unpractical.

When we consider the difference between he control simulations with 32 and 8 sample points, we see that the error did not grow substantially. The model works relatively well with few sample points. This means that we do not need to increase the number of sample points artificially by re-using them for multiple time steps. The results are mostly better if a lower number of sample points is evaluated every time step.

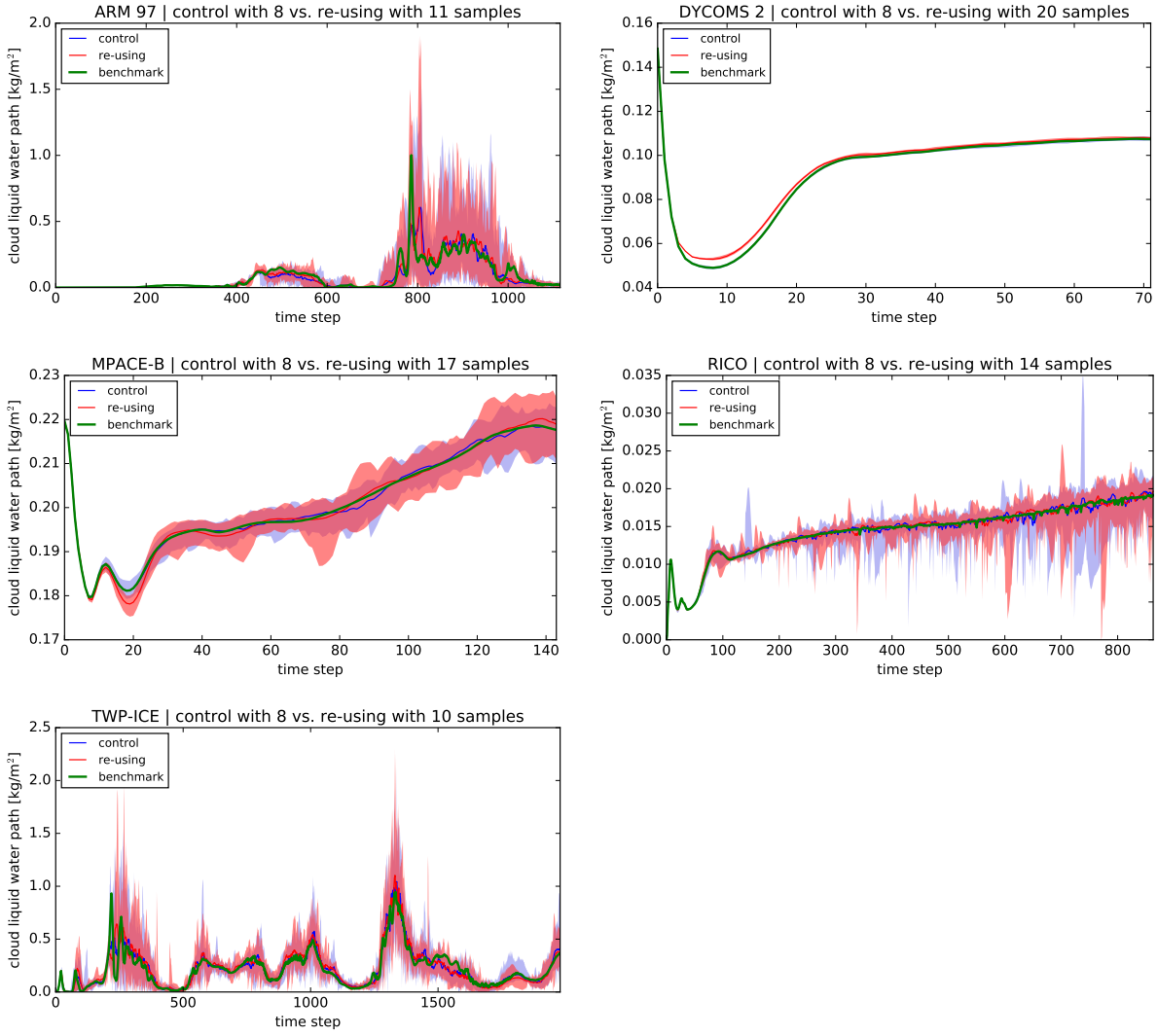

Figure 3.6: Results for LWP for the re-using algorithm and equal computation time.

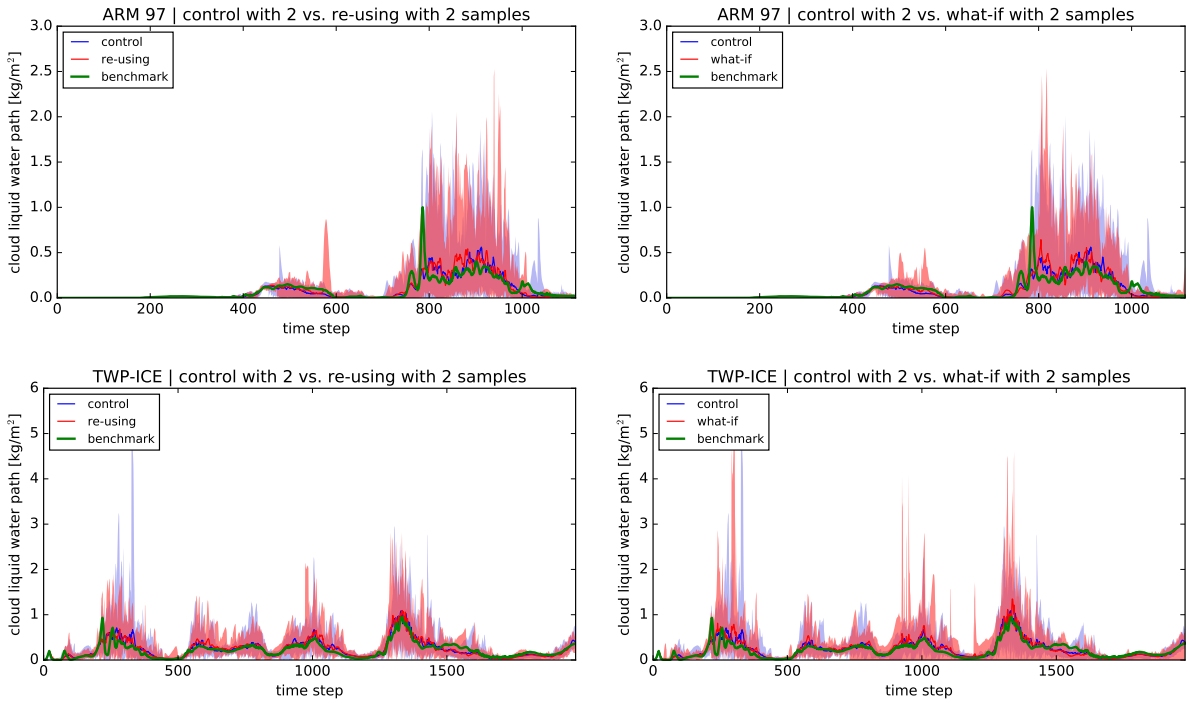

Figure 3.7: Solutions for LWP computed with two sample points.

3.4 Simulations with minimal sample points

We have shown in Sections 3.2 and 3.3 that neither the what-if sampling nor the un-weighted re-using can improve simulation results for fixed computation costs. The promising idea of re-weighting old sample points turned out to be too sensitive to bad importance sampling and too computationally expensive due to the computation of the density functions. The simple re-using method benefits from the negligible overhead and is slightly better, but real improvements could not be achieved. The ultimate goal for optimizing SILHS is to reduce the costs of the microphysics to one call per time step. This would make the model computationally competitive to simple subgrid parametrizations of existing models and increase the usability. So we want to set the number of sample points to the minimum and see how the methods perform. Calculations with one sample point are not possible with SILHS, because co-variance terms would be zero, or not computable when Bessel's correction is used. Therefore, we take two sample points for the control, what-if and re-use simulation and see

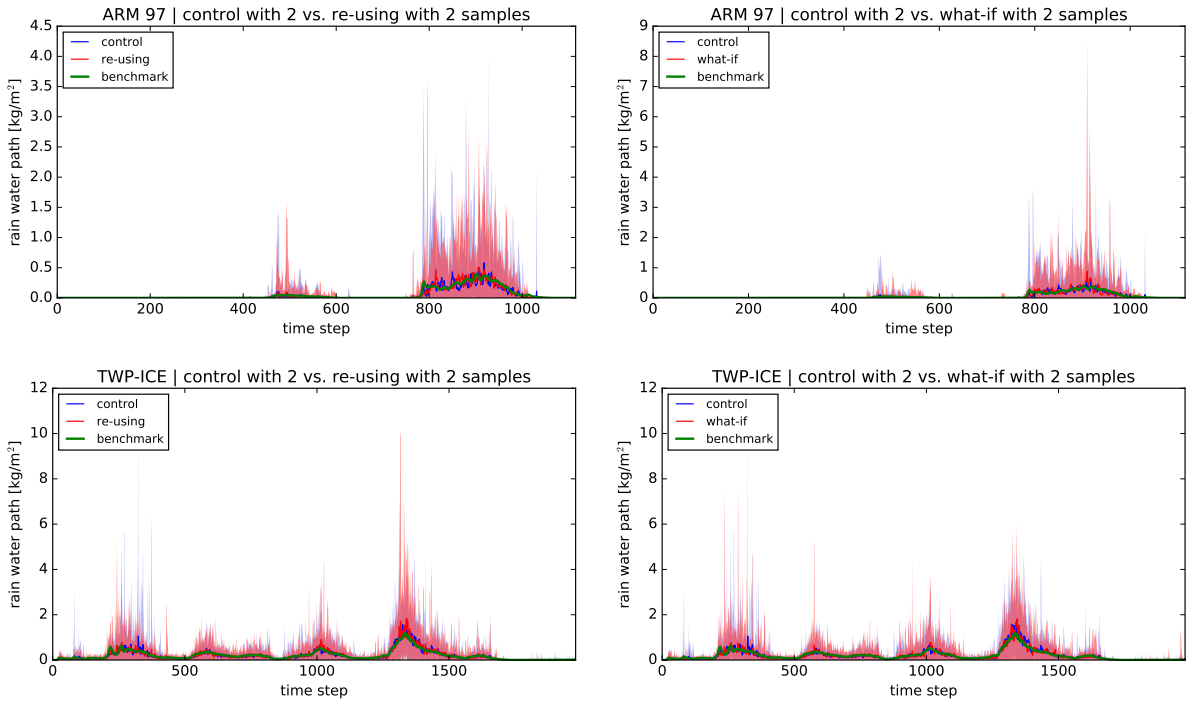

Figure 3.8: Solutions for RWP computed with two sample points.

if the speedup makes up for the additional noise in the solution. The spread of the solutions for LWP and RWP are shown in Figures 3.7 and 3.8. We focus on the interesting cases ARM 97 and TWP-ICE, the figures for the other cases are given in Appendix B. The means of all ensembles are still very good. Neither what-if sampling nor the re-using method show a significantly larger spread than the control simulation.

Tables 3.5 and 3.6 show the errors and relative computation times for the simulations. The errors of the adaptive methods are just a little larger, mostly the errors increase by less than 20%. On average the what-if algorithm is slightly better than re-using, but the computation time of what-if sampling is too high again. So the re-using algorithm would be used to reduce the costs. The number of calls to the microphysics can be reduced by 20% for ARM 97, 56% for DYCOMS 2, 40% for MPACE-B and RICO, and 24% for TWP-ICE. The goal of reducing the microphysics cost to one call per time step could only be accomplished once.

	avg. error LWP			*avg. error RWP*		
case	*control*	*re-use*	*what-if*	*control*	*re-use*	*what-if*
ARM 97	5.85e-02	5.86e-02	5.82e-02	4.20e-02	4.25e-02	4.50e-02
DYCOMS 2	6.89e-04	1.73e-03	1.72e-03	1.79e-04	3.18e-04	2.77e-04
MPACE-B	4.64e-03	6.33e-03	6.66e-03	4.83e-04	5.68e-04	5.44e-04
RICO	1.31e-03	1.29e-03	1.10e-03	1.85e-03	1.88e-03	1.74e-03
TWP-ICE	1.08e-01	1.11e-01	1.20e-01	1.30e-01	1.42e-01	1.42e-01

Table 3.5: Average error of simulations with two sample points.

	physics costs		*total costs*	
case	*re-use*	*what-if*	*re-use*	*what-if*
ARM 97	80 %	86 %	80 %	106 %
DYCOMS 2	44 %	46 %	44 %	55 %
MPACE-B	60 %	67 %	60 %	84 %
RICO	60 %	94 %	60 %	132 %
TWP-ICE	76 %	81 %	76 %	100 %

Table 3.6: Computation time of simulations with two sample points in comparison to the control simulation.

The re-using and what-if sampling algorithm are not significantly worse than SILHS with 2 sample points. Considering the computation time, only the re-using algorithm is interesting for application. If the simulation results of SILHS with two sample points are good enough, the results of the re-using method are also good enough, since they did not worsen significantly. However, it depends on the field of application if such strong noise is acceptable or not.

3.5 Other approaches for computation time reduction

3.5.1 Redrawing a subset of samples

We have seen that drawing new sample points at every time step is better than using more sample points for more than one time step. However, we could just draw a few sample

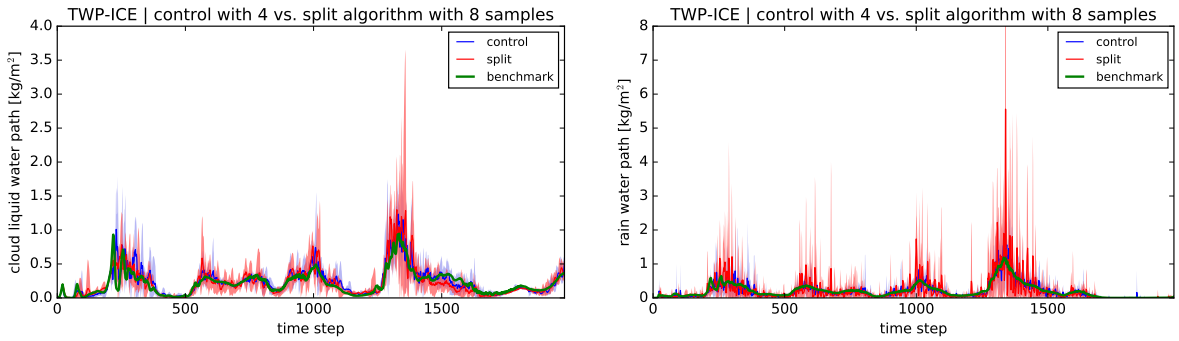

Figure 3.9: Solutions for LWP and RWP computed with the algorithm that updates half of the sample points every time step.

points at each time step and include the sample of the previous time steps for the average calculation. The problem is that the expensive re-weighting is required for this approach to converge. For this test we ignore that the old sample have to be re-weighted and use the old weights as they are. We applied this algorithm to the TWP-ICE case. The simulation uses a total of 8 sample points, one half of them is updated every time step. The control simulation uses 4 sample point, which requires the same computation time.

Figure 3.9 shows the results of 5 ensemble simulations. The first 500 time steps for the LWP are solved significantly better with the new algorithm, after that no real advantages are visible. The RWP is very noisy for this method, which illustrates that additional sample points do not help, if they are not weighted correctly. The assumptions for this algorithm are so vague. The average errors for LWP and RWP are 8.56×10^{-2} and 1.03×10^{-1} for the control and 1.06×10^{-1} and 1.86×10^{-1} for the new algorithm, respectively. Therefore, the simple implementation of this algorithm does not suggest that this approach is more promising than simple re-using. However, re-weighting of the old half of the samples and adding adaptive criteria could make this approach more competitive. The problem of this method is that the module SILHS was not designed to draw a subset of samples, such that large code refactoring would be necessary for an efficient and well designed implementation.

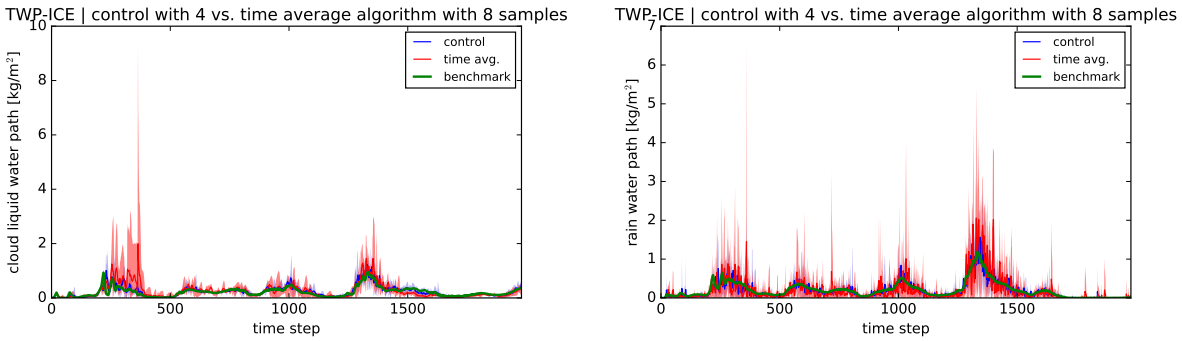

Figure 3.10: Solutions for LWP and RWP computed with the algorithm that uses a weighted time average of the tendencies.

3.5.2 Time averages

Instead of working with the sample points, we could also use the resulting averages of the last few time steps to estimate the new tendencies at the current time step. In this test we compute a weighted average of the last four time steps, which will be weighted by 50%, 25%, 15%, and 10%. The average will not predict the future conditions, but it will describe the overall conditions relatively well. So the maximal error of this method is expected to be smaller, while the average error is expected to increase. We used the TWP-ICE case for this test and set the number of sample points to 8. After four time steps were computed, the microphysics will be omitted every other time step and the new tendencies are estimated as described above. So a usual simulation with 4 sample points is about equally computational expensive.

Figure 3.10 shows the average, min and max solutions of 5 simulations for LWP and RWP. Our assumption was that large errors will be prevented by the averaging, but the results for the individual time steps can be too noisy from time to time. In the end, the bad estimates even have a larger impact than before and reduce the quality of the simulation by creating even more outliers. The average errors of this method are 1.35×10^{-1} for LWP and 1.70×10^{-1} for RWP, which is again much more than for the control simulation.

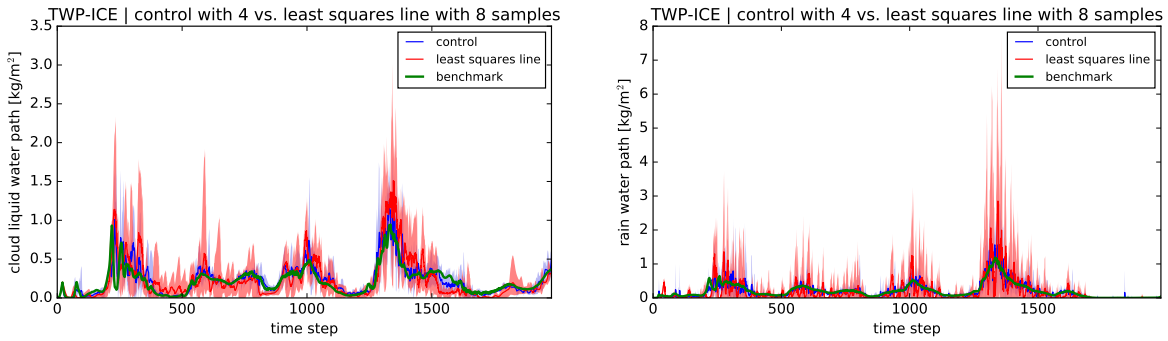

Figure 3.11: Solutions for LWP and RWP computed with the least squares line approximation of the tendencies.

3.5.3 Polynomial extrapolation

An idea was to use a linear multistep methods like the Adams-Bashforth method, which solves an ordinary differential equation (ODE) by approximating the first derivative with a polynomial. In this case, we use the solutions of the tendencies of the last time steps to fit a polynomial to the data, which is used to extrapolate the new tendencies. Since the estimates oscillate around the true grid average, a polynomial though all points will most likely be too noisy and not close to the true solution. Therefore, we decided to compute the least squares solution for the data. In this example we consider a straight line that is defined by the least squares approximation of the last four time steps, because the computation is relatively cheap. We use the same design as for the last method. After reference time steps have been computed, we will skip the microphysics every other time step and estimate the tendencies with the least squares line. Figure 3.11 shows the solutions for this test. We see that the least squares approximation can remove larger spikes relatively well, but the overall accuracy is not specially good. The average errors for LWP and RWP for the control simulations are again 8.56×10^{-2} and 1.03×10^{-1} and the average errors of the algorithm are $1.55\times10^{-1}$and 2.38×10^{-1}. So, the average errors are higher for this algorithm than for the previous two. A problem is that the data of 20 minutes is used for the approximation and all data points are weighted equally. The general evolution of the time tendency, given by the least squares approximation, seems to be a relatively bad estimate for the current tendency.

Chapter 4

Discussion

4.1 Discussion and conclusion

The goal of this thesis was to develop an adaptive Monte Carlo sampling algorithm for the computation of grid average time tendencies.

In the beginning, the basic concepts of Monte Carlo sampling and importance sampling were introduced. These concepts were applied to develop the what-if sampling algorithm that re-uses old sample points and their function values. This is done by re-weighting according to the ratio of the PDFs. Examples for importance sampling and the what-if sampling illustrated the advantages and disadvantage of the methods. Huge gains in accuracy of the estimator are possible, but if the source PDF does not fit to the conditions the opposite effect is also possible. Therefore, adaptive criteria were developed to filter out the outliers and give a stable adaptive what-if sampling algorithm. The mean shift, the effect sample size, the age of the sample points and the conditions at the most important vertical layer were taken into account to decide if what-if sampling should be applied or if new samples should be drawn.

The convergence test has shown that the average tendencies converge to the expected solution. However, the atmospheric conditions have a large impact on the quality of the re-weighting. Good conditions are present when precipitation decreases or when the atmospheric conditions are more or less uniform.

The real case studies have revealed that the what-if sampling algorithm is not cost efficient in comparison to the standard SILHS implementation. It is more accurate to use fewer sample points that are updated every time step than to use more sample points that are

re-weighted from time to time. The major disadvantages are the high overhead of the re-weighting and the extra noise caused by anti-importance sampling. The overhead comes from the computation of the density function of the multidimensional normal distribution. A major problem of the weight calculation is that CLUBB's PDF is so high dimensional. The means of twelve variates can shift, which often causes huge or zero weights. The best possible results for what-if sampling in this context were similar results for the same computation time.

We tried to make the what-if sampling algorithm more applicable by omitting the re-weighting. Therefore, we assume that the tendencies do not change significantly between two time steps. This reduced the overhead to zero and allowed faster simulations with even more sample points. However, this algorithm is also worse than standard SILHS when the same computation time is used. One possible application for the re-using algorithm is the computation of simulations with very few sample points. When only two samples are used, the results do not degrade strongly when re-using or what-if sampling is applied, but the costs can be reduced by 20-50%.

In conclusion, the sampling of SILHS gives relatively good estimates for the tendencies even if few samples are used. This makes it difficult for heuristic algorithms, like the re-using, to improve the results. The what-if sampling method is not suitable for this context because the distribution of CLUBB makes re-weighting too difficult. However, there probably are other applications that can profit from the what-if sampling algorithm.

Bibliography

Ackerman, T. P., A. D. Del Genio, G. M. McFarquhar, R. G. Ellingson, P. J. Lamb, R. A. Ferrare, C. N. Long, S. A. Klein, and J. Verlinde, 2004: Atmospheric Radiation Measurement program science plan. http://www.arm.gov/science.

American Meteorological Society, 2017: *Glossary of Meteorology*. [Available online at http://glossary.ametsoc.org/wiki/Turbulent_flux].

Bogenschutz, P. A., A. Gettelman, H. Morrison, V. E. Larson, C. Craig, and D. P. Schanen, 2013: Higher-order turbulence closure and its impact on climate simulations in the Community Atmosphere Model. *J. Climate*, **26**, 9655–9676.

Forbes, R., 2015: Sub-grid cloud parameterization. [Available online at http://www.dtcenter.org/events/workshops15/mm_phys_15/presentations].

Golaz, J.-C., V. E. Larson, and W. R. Cotton, 2002: A PDF-based model for boundary layer clouds. Part I: Method and model description. *J. Atmos. Sci.*, **59**, 3540–3551.

May, P. T., J. H. Mather, G. Vaughan, K. N. Bower, C. Jakob, G. M. McFarquhar, and G. G. Mace, 2008: The Tropical Warm Pool International Cloud Experiment. *Bull. Amer. Meteor. Soc.*, **89**, 629–645.

Morrison, H., J. A. Curry, and V. I. Khvorostyanov, 2005: A new double-moment microphysics parameterization for application in cloud and climate models. Part I: Description. *J. Atmos. Sci.*, **62**, 1665–1677.

Owen, A. B., 2003: Quasi-Monte Carlo techniques. *Siggraph 2003, Course 44*, San Diego, CA, Association for Computing Machinery.

Owen, A. B., 2013: *Monte Carlo theory, methods and examples*.

Rauber, R. M., B. Stevens, H. T. Ochs, C. Knight, B. A. Albrecht, A. M. Blyth, C. W. Fairall, J. B. Jensen, S. G. Lasher-Trapp, O. L. Mayol-Bracero, G. Vali, J. R. Anderson, B. A. Baker, A. R. Bandy, E. Burnet, J.-L. Brenguier, W. A. Brewer, P. R. A. Brown, P. Chuang, W. R. Cotton, L. D. Girolamo, B. Geerts, H. Gerber, S. Göke, L. Gomes, B. G. Heikes, J. G. Hudson, P. Kollias, R. P. Lawson, S. K. Krueger, D. H. Lenschow, L. Nuijens, D. W. O'Sullivan, R. A. Rilling, D. C. Rogers, A. P. Siebesma, E. Snodgrass, J. L. Stith, D. C. Thornton, S. Tucker, C. H. Twohy, , P. Zuidema, K. R. Sperber, and D. E. Waliser, 2007: Rain in shallow cumulus over the ocean: The RICO campaign. *Bull. Amer. Meteor. Soc.*, **88**, 1912 – 1928.

Stevens, B., D. H. Lenschow, G. Vali, H. Gerber, A. Bandy, B. Blomquist, J.-L. Brenguier, C. S. Bretherton, F. Burnet, T. Campos, S. Chai, I. Faloona, D. Friesen, S. Haimov, K. Laursen, D. K. Lilly, S. M. Loehrer, S. P. Malinowski, B. Morley, M. D. Petters, D. C. Rogers, L. Russell, V. Savic-Jovcic, J. R. Snider, D. Straub, M. J. Szumowski, H. Takagi, D. C. Thornton, M. Tschudi, C. Twohy, M. Wetzel, and M. C. Van Zanten, 2003: Dynamics and chemistry of marine stratocumulus — DYCOMS-II. *Bull. Amer. Meteor. Soc.*, **84**, 579–593.

Thayer-Calder, K., A. Gettelman, C. Craig, S. Goldhaber, P. A. Bogenschutz, C.-C. Chen, H. Morrison, J. Höft, E. Raut, B. M. Griffin, J. K. Weber, V. E. Larson, M. C. Wyant, M. Wang, Z. Guo, and S. J. Ghan, 2015: A unified parameterization of clouds and turbulence using CLUBB and subcolumns in the Community Atmosphere Model. *Geosci. Model Dev.*, **8**, 3801–3821.

Verlinde, J., J. Y. Harrington, and Co-Authors, 2007: The mixed-phase Arctic cloud experiment. *Bull. Amer. Meteor. Soc.*, **88**, 205–221.

Appendix A

Convergence

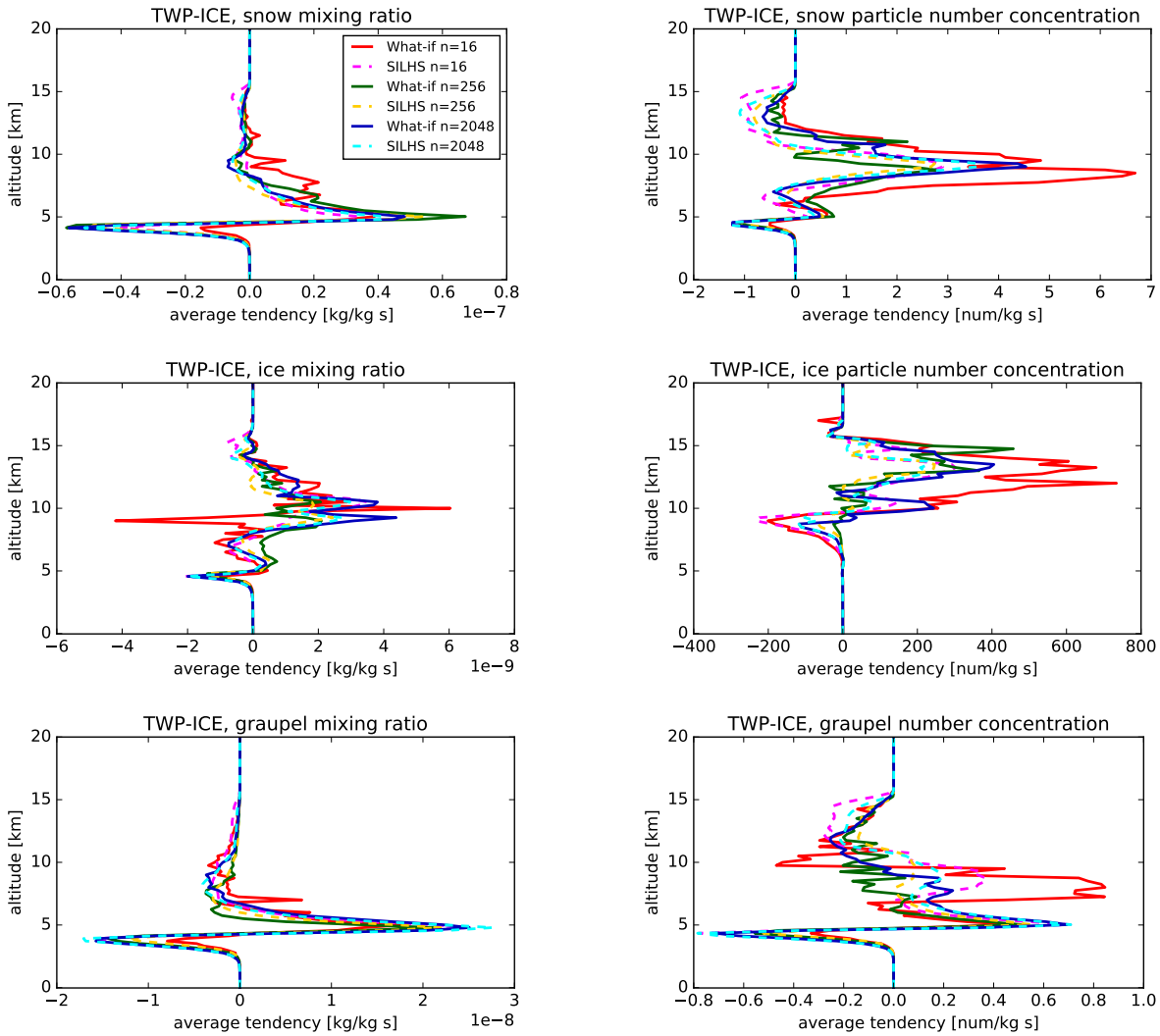

Figure A.1: Average tendencies for snow, ice and graupel mixing ratios and number concentrations.

Appendix B

Results

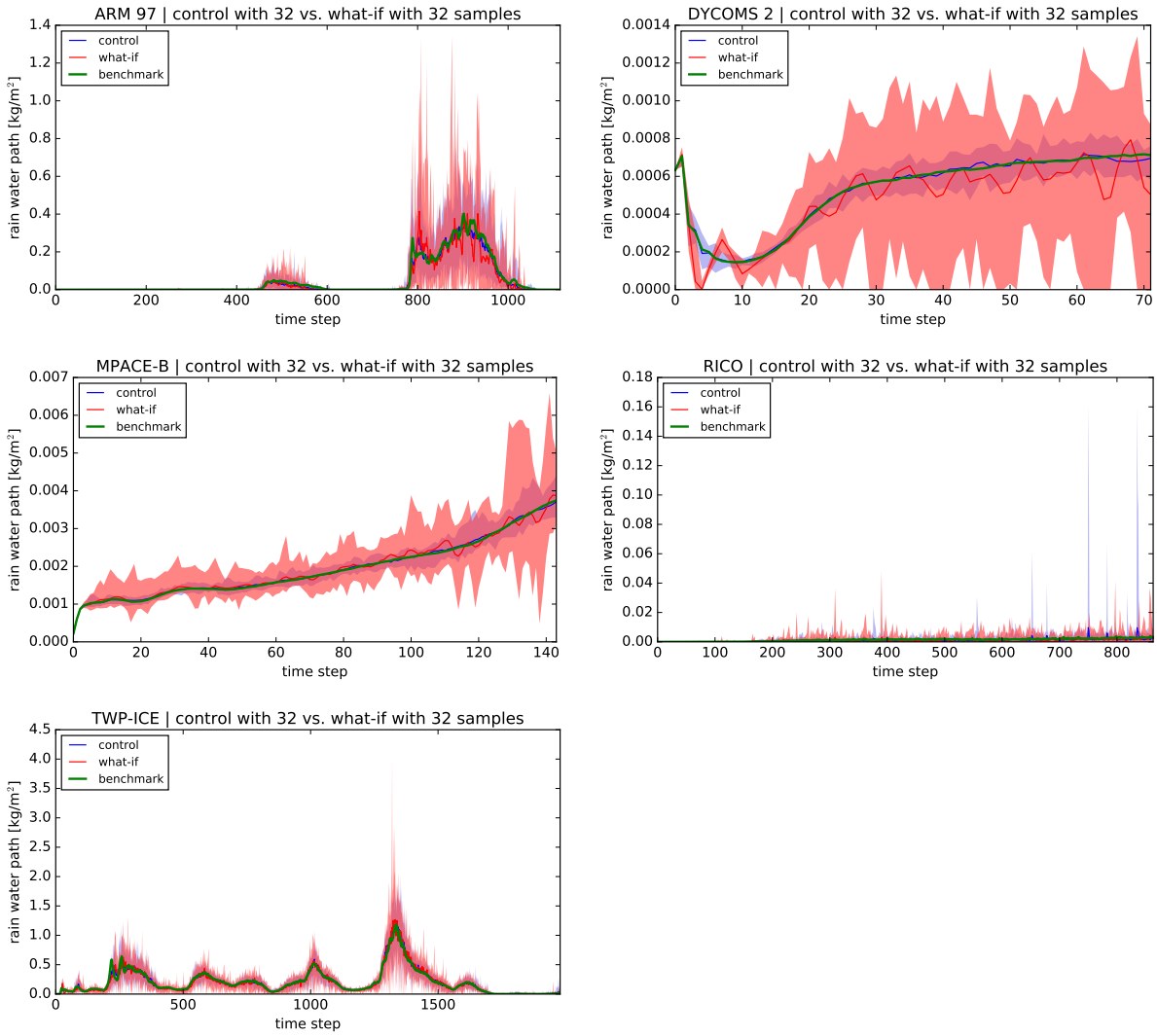

Figure B.1: Results for RWP for what-if sampling with 32 sample points.

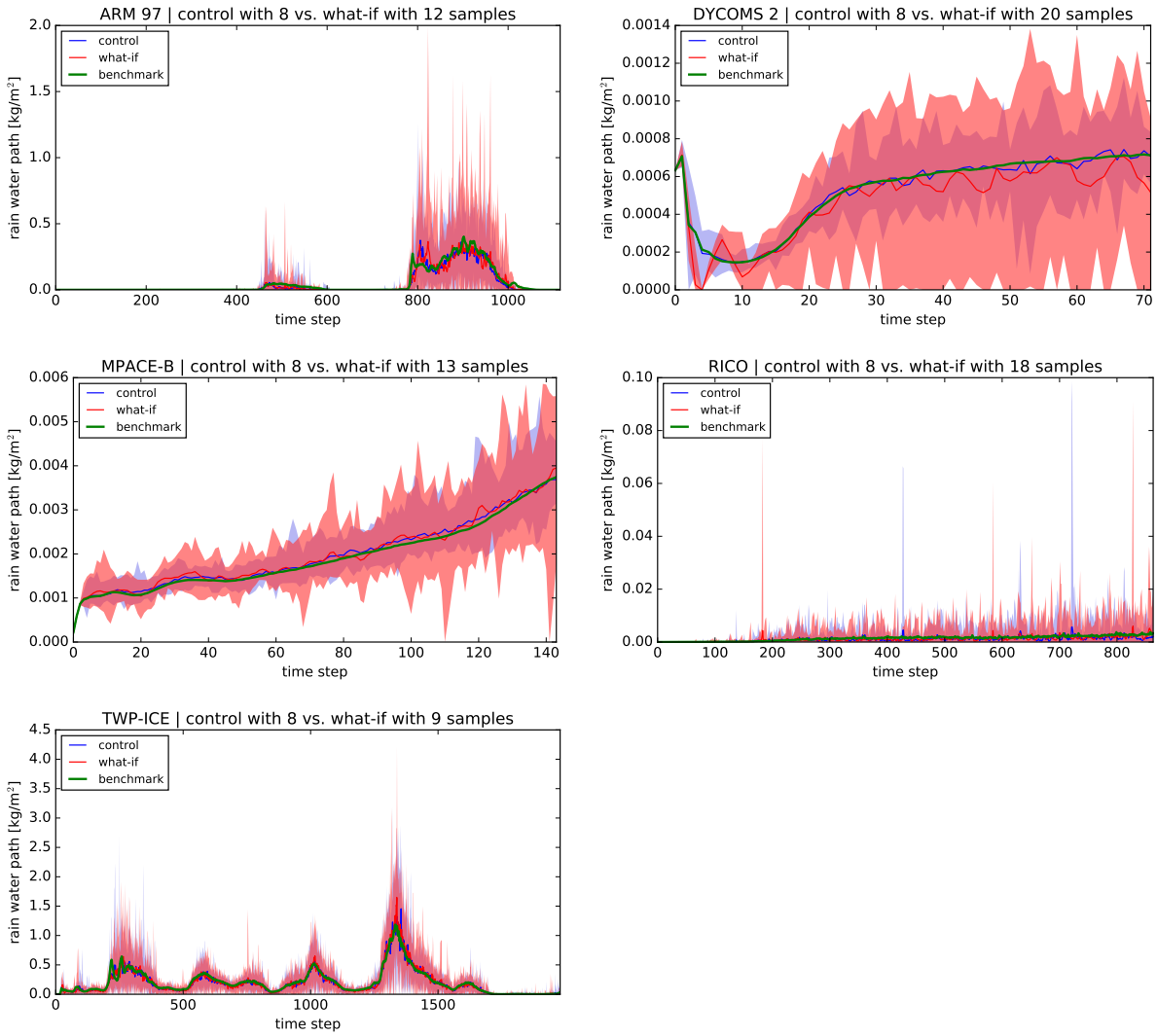

Figure B.2: Results for RWP for what-if sampling and equal computation time.

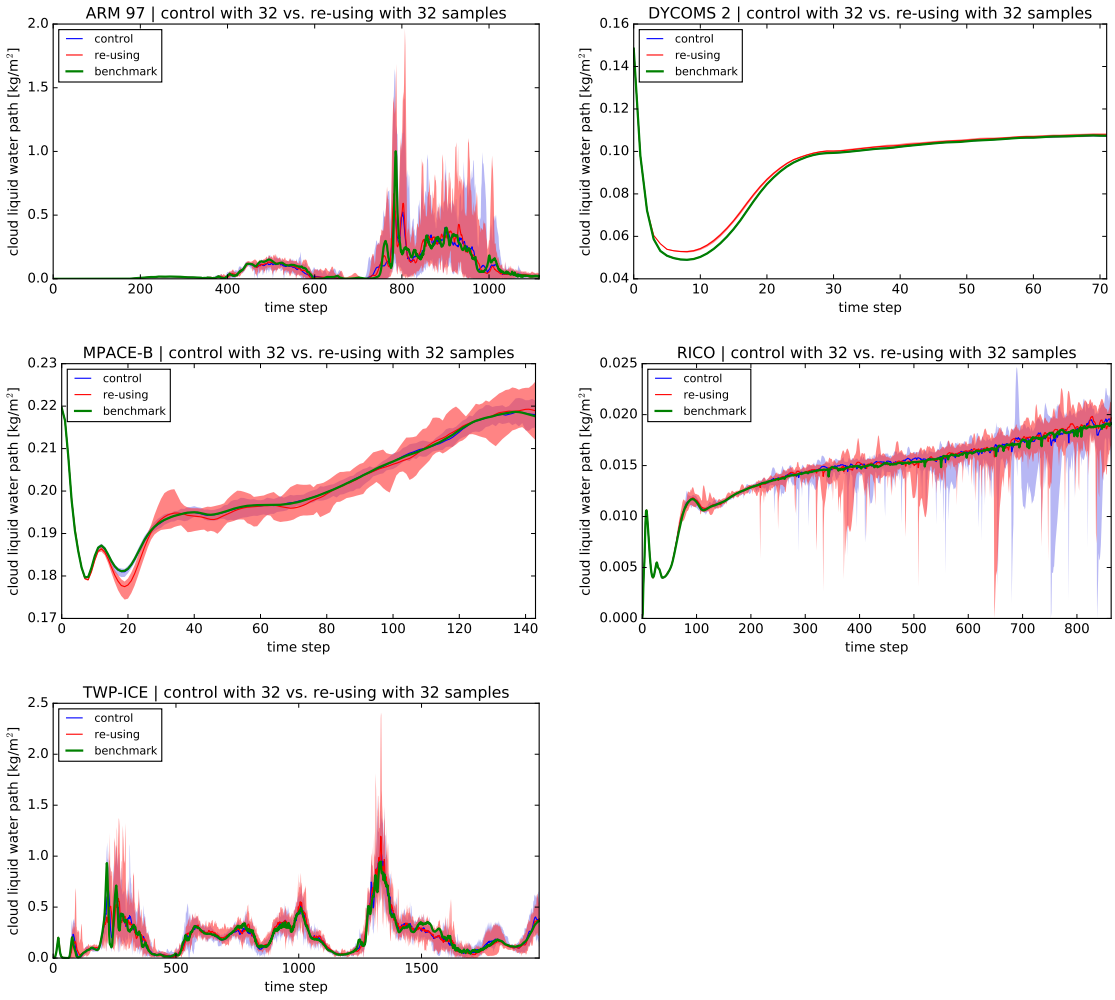

Figure B.3: Results for RWP for the re-using algorithm with 32 sample points.

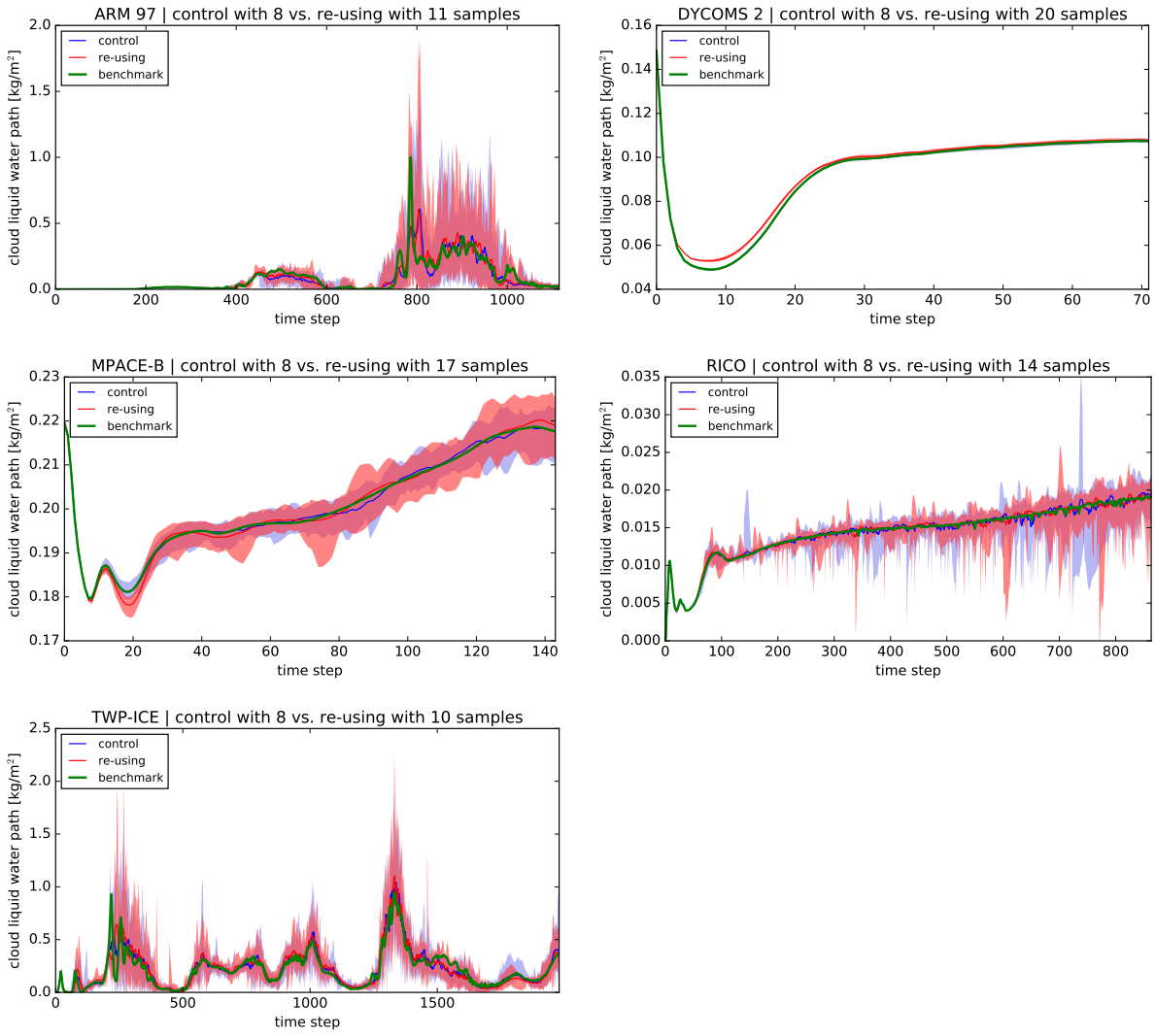

Figure B.4: Results for RWP for the re-using algorithm and equal computation time.

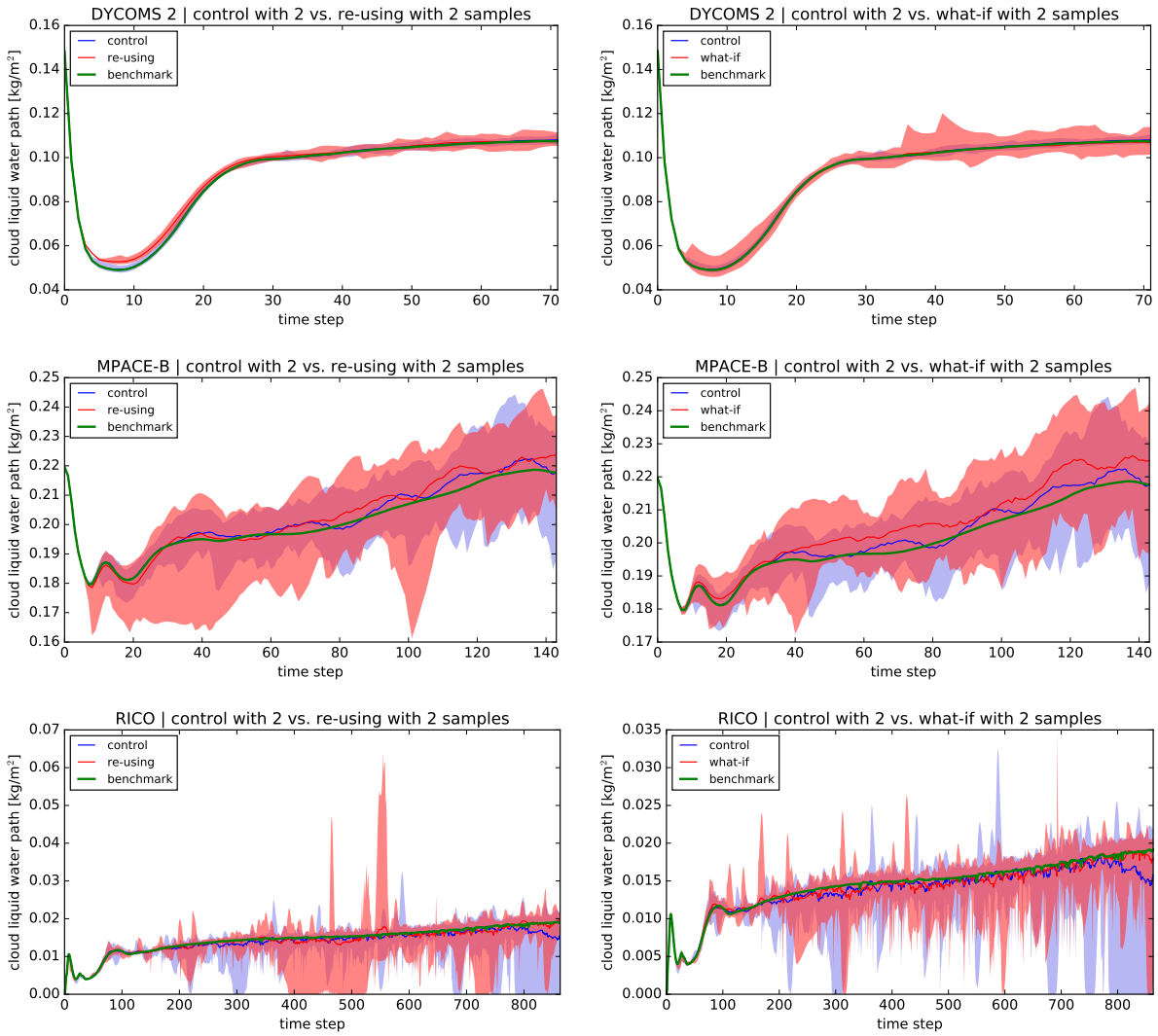

Figure B.5: Results for LWP for the relatively easy cases with two sample points.

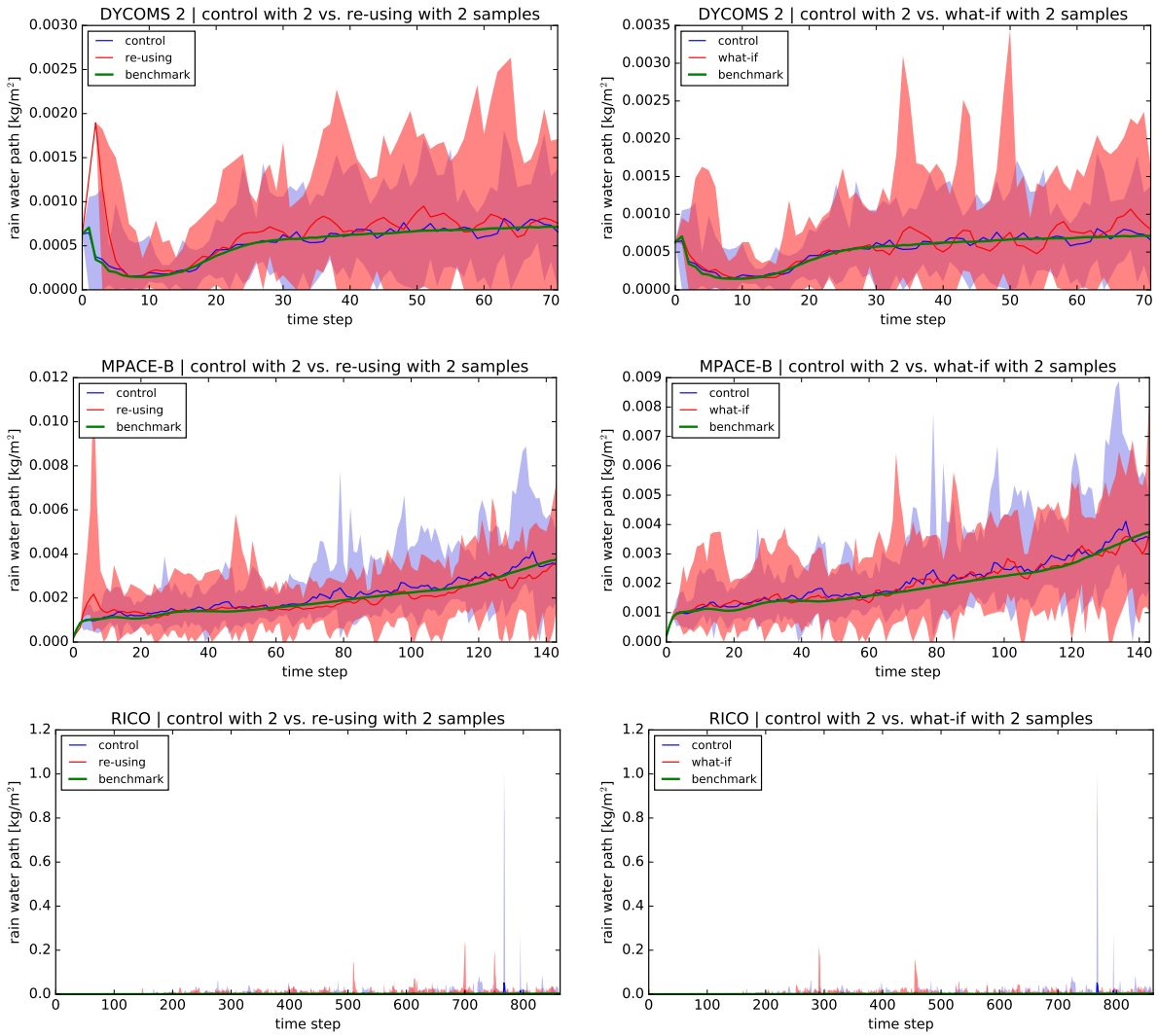

Figure B.6: Results for RWP for the relatively easy cases with two sample points.

Appendix C

Performance tuning

The reweighting algorithm takes relatively much computation time due to the required inverse of the covariance matrix and the evaluation of the exponential function. It has to be significantly faster than the standard SILHS implementation to make the algorithm useful.

The correlations of the variates are prescribed and constant in time. The only difference is that a different correlation matrix is used in precipitating and dry conditions. With help of a Cholesky decomposition, we can compute the inverse of the correlation matrix beforehand to save computation time. The inverse of the covariance matrix can be calculated by a simple multiplication with the standard deviations. However, we have the problem that only a subset of variates is used at every time step. For nine optimal variates and two sets of correlations, we end up with 1024 combinations of the variates. For most test cases, the simulations are not much longer than 2000 time steps, so precomputing the inverse matrices is not much cheaper than computing them on the fly. Therefore, the what-if module computes the necessary Cholesky matrix when it is needed. After that the invers of the correlation matrix is stored and can be loaded when the same combination of variates is used again. Roughly 50 of the 1024 combinations are needed for difficult test cases like TWP-ICE.

The model's grid is usually much larger than the actual event. The RICO case for example extends to about 6 km altitude, while the standard grid is 30 km high. The re-weighting of tiny microphysical tendencies in regions without significant concentrations of hydrometeors is expensive but does not influence the quality of the results. Before microphysical tendencies are re-weighted the magnitude of the tendencies is analyzed. Layers without relevant processes will just use the default weights.