Geosci. Model Dev. Discuss., 7, C1749–C1766, 2014
www.geosci-model-dev-discuss.net/7/C1749/2014/

Geoscientific
Model Development
Discussions
Open Access

Interactive comment on "Improved routines to model the ocean carbonate system: mocsy 1.0" *by* J. C. Orr and J.-M Epitalon

J. C. Orr and J.-M Epitalon

james.orr@lsce.ipsl.fr

Received and published: 15 September 2014

Response to Referee Comment by Guy Munhoven

We are indebted to Dr. Munhoven for his multitude of comments, many of a highly technical nature, which will lead to substantial improvements in the mocsy code and in the revised manuscript. His comments are repeated below in gray, while our responses follow in black.

1 General comments

1.1 Appreciation of the manuscript

In this paper, J. C. Orr and J.-M. Epitalon describe a new Fortran 95 library to model the ocean carbonate system, MOCSY. MOCSY updates and extends the widely used codes from the Ocean Carbon Cycle Model Intercomparison Project, OCMIP (Orr et al., 2000). It extends the choice of stoichiometric constant parameterizations, to comply, as far as possible with the recommendations for best practices (Dickson et al., 2007). Deviations from best practices are clearly stated and entirely justified. MOCSY furthermore provides more complete speciation calculations than the OCMIP codes.

MOCSY also offers its users the possibility to take into account developments that got published after the best practices. These include the new boron-to-salinity ratio from Lee et al. (2010) and more recent parameterizations for the two dissociation constants of carbonic acid. It furthermore goes beyond the recommendations for best practices, by also providing the necessary formulations for pressure correction (Dickson et al. (2007) does not include these since its focus is on the analytical aspects of carbonate system measurements, generally made at atmospheric pressure).

Finally, the paper also presents a quantitative assessment of several approximations often made in models: (1) density is constant; (2) the approximation of in-situ temperature by potential temperature (the former being, strictly speaking, the temperature actually required for the chemistry calculations, while the latter is what models provide); (3) the equivalence between depths in meters and pressure in dbar; (4) nutrient contributions to alkalinity (i.e., from the silicic and phosphoric acid systems) are negligible.

The paper is generally well written. The history and motivations behind the development of MOCSY are informative. In a few instances, the text is not entirely precise—this can, however, be easily fixed. The selection of material to demonstrate the potential of the package is good. Figures are of good quality, but might be improved for a few details (coordinate axes extents).

The model description is somewhat short. While more or less all of the thermodynamical ingredients are described in full detail, close to nothing is provided regarding the numerical methods applied to actually perform the pH calculations. Above all, the little that is provided is contradictory. Geoscientific Model Development papers are expected to present a complete and detailed description of the models. A minimum of technical details about the methods and algorithms used should be given and limitations should also be discussed (see “Manuscript Types” in the “Submission” guidelines). The presentation and discussion of the results contain, unfortunately, a few inaccuracies (see detailed comments below). These do, however, not impinge on the overall conclusions and can be easily corrected.

In the revised manuscript, we will elaborate on the method used to compute pH and we will do our best to remedy any inaccuracies.

The “Code availability” section in the paper and the code distribution channel are exemplary. The code is distributed under the MIT license, and can be easily obtained. The included test case compiled and worked out of the box as expected for me.

I am confident that the authors will be able to address all of the shortcomings mentioned. Accordingly, I recommend to publish this paper after a revision of the text.

Much appreciated.

1.2 Discussion

1.2.1 Constant, variable, or no density correction at all?

Regarding issue (1) above, I expect that applying a constant or even no density correction at all should not make much of a difference in the calculation results! This may seem counterintuitive, but, it is sufficient to notice the following;

1. all of the equilibrium constants (except for K_W) carry the units of the proton concentration—for all other concentrations that appear in the equilibrium relationships, any unit conversion factors cancel out;

2. in the alkalinity-pH equation, again, only the ratios of the different total concentrations (total dissolved inorganic carbon, C_T, total borate, B_T, sulfate, S_T, etc. $\ldots$) to total alkalinity, A_T, are relevant and, as long as all of these are expressed relative to the same reference framework (volume or mass), any density conversion factors will cancel out; the only term in the alkalinity-pH equation that might be affected by an imprecise density will be A_W/A_T, where the water self-ionization alkalinity, A_W, directly depends on the proton concentration units. That ratio is, however, only of the order of a per mil and an error of the order of a percent in this actual ratio should not make any significant difference in the calculated proton concentration.

Whatever the units of the total input concentrations, the speciation results will not be affected to any significant extent, since the speciation routine only determines, e.g., what fraction of C_T is CO_2, HCO_3- or CO_3^{2-}. Only the calculated proton concentration will always come out in the units of the equilibrium constants (generally mol/kg-soln) and it would be best not to convert it to a volumetric reference at all.

Thank you for this clear explanation, which we will use to improve the revised manuscript. It is consistent with the finding in the Discussion paper that the constant density approximation does not produce significant errors, unless a different constant density is used before (to convert the model input from mol m^{-3} to mol kg^{-1}) and after (when converting back to model units of mol m^{-3}).

Using a different density conversion factor during subsequent model-data comparison than the one that was possibly used for the speciation calculation should not make any significant difference. The model-data comparison must anyway be carried out consistently at the density of the data samples.

Our tests indicate that using a different density conversion factor matters. But we will revisit this issue and detail our findings as to why and how much things differ if we use a different constant density for concentration conversions before and after the carbonate chemistry calculations.

1.2.2 Depth-pressure conversion: how significant are the errors that we try to avoid?

Regarding issue (3) above, Lewis and Wallace (1998) already noted that depth expressed in meters and pressure expressed in dbar differ by only 3% at 10,000 dbar and less at lower pressures, which is well within the uncertainties of the pressure effects on the equilibrium constants. To my best knowledge, there has not been any progress in reducing the uncertainties of the pressure correction coefficients since the compilation of Millero (1995) and the reviewing efforts of Lewis and Wallace (1998): the currently used coefficients are more than 30 years old; some of them remain are merely expert guesses – the coefficient for the first dissociation constant of silicic acid is estimated from that of boric acid (Millero, 1995).

We agree that there is a dire need for studies devoted to better estimate the pressure adjustments needed for the equilibrium constants. Although this is not something that mocsy or other carbonate chemistry packages can answer, it is also not an excuse to be sloppy in our calculations. We do not think that it is a huge effort to make the pressure corrections as they were intended to be made. We do this in mocsy as do other public packages (CO2SYS, seacarb, etc).

Accordingly one may ask whether the tiny improvement resulting from the more accurate depth-to-pressure conversion is really worth the effort, since the resulting changes do not have any measurable significance.

Yes the uncertainties in the basic pressure correction are large and the difference between meters and decibars small. However, the latter is a systematic bias that is easy to correct for and requires insignificant additional computational resources. The mocsy package allows the user to choose whether or not to make this correction. It does not impose it. In the Discussion paper we show that it is small error. In the revised manuscript we will emphasize this point further, but we will still allow mocsy users to decide on the best approach for them.

Furthermore, the common practical application of the pressure correction of the equilibrium constants leaves, in my opinion, an unpleasant aftertaste of inconsistency: for any chosen work pH scale, to adjust the value of an equilibrium constant calculated at atmospheric pressure to any other given pressure, Lewis and Wallace (1998) recommend to first convert it to the seawater (SWS) or the NBS scales, then apply the pressure correction, and finally convert the result back to the selected work pH scale (using the pH scale conversion factor at the final pressure). Since NBS scales are nowadays rarely used in seawater chemistry, the pressure correction is generally made on the SWS scale. Results will nevertheless most probably be different depending on whether the SWS or the NBS pathways are chosen. This protocol must be followed for all equilibrium constants, except for the dissociation of hydrogen sulfate and fluoric acid, which should be pressure-adjusted on the free scale. These are indeed required for the pH scale conversions, and the common reference in pH scale conversions is the free pH scale.

The cited text from Lewis and Wallace (1998) lends itself to misinterpretation. In fact, for a given constant, there is no choice for the pH scale upon which the pressure correction is made. For K_S and K_F the pressure correction should be made on the free scale. For the other constants (except K_0, K_A, and K_C, which are independent of pH scale) the pressure correction must be made on the seawater scale. Our two previous responses also address this criticism as does our next response.

While I see the usefulness of a well-defined protocol for performing these pressure adjustments, we are now in the somewhat surrealist situation where the best practices recommend the use of equilibrium constants on the total pH scale, these constants need to be temporarily converted to the SWS scale for pressure correction, and back to the total scale, except for two of them, that need to stick to the free scale, although the pressure correction coefficients are certainly not precise enough to distinguish between the different scales. Why not carry out the pressure correction on the free scale and gain at least in consistency?

Today, all seven public packages that compute ocean carbonate chemistry that were compared by Orr et al. (2014) use the same approach to make pressure adjustments to the constants. Our mocsy package is one of those. The only exception is the csys package, which made pressure adjustments on the total scale instead of the seawater scale. But the latest version of csys, recently revised, now offers the option to follow the standard procedure outlined by Lewis and Wallace (1998). We see no good reason to change this approach in mocsy. Moreover, converting constants intended to be on the seawater scale to the free scale before making pressure corrections is a more substantial adjustment that would lead to significant systematic biases in their pressure-corrections.

1.2.3 Nutrient alkalinity: how significant are the differences between the results?

Although, once again, the assessment and the discussion of the importance of nutrient related acid systems in the alkalinity composition is very interesting, I am not convinced about the significance of the consequences of including or neglecting them. As far as I can see, the quality of even our most advanced global model results is still far from sufficient to make the observed differences relevant. Locally in the deep-sea, inter-model differences in the calculated carbonate ion profiles remain far larger than the quoted 1.6 μmol/kg. A polynomial pH-solver based upon practical alkalinity may easily be an order of magnitude faster (even if safe-guarded) than a full solver required with the complete representation of AT.

As pointed out in our Discussion paper, the neglect of nutrient alkalinity leads to a systematic bias of up to -6 μatm in pCO$_2$ for the Southern Ocean zonal mean. We think that is substantial, even if it comes from a model. A bias of -6 μatm might seem small, but a shift of that magnitude is enough in many models to incorrectly assign the Southern Ocean as a sink rather than a source. We recall also that a global air-sea CO$_2$ flux of 2 Pg C yr^{-1} (roughly the ocean's current uptake of anthropogenic CO$_2$) results from only a 4 μatm difference between atmospheric and oceanic pCO$_2$. In the late 1990s there was a discussion about whether or not to include phosphoric and silicic acid systems in the alkalinity equation for the protocols of the model simulations made

as part of the Ocean Carbon Cycle Model Intercomparison Project (OCMIP, Phase 2). It was agreed then to include both to avoid the systematic biases mentioned above.

Hence, we think it is inappropriate to neglect nutrient alkalinity in model simulations even if it costs more. And we suspect that the extra cost must remain a very small fraction of the total time needed to run most large-scale ocean biogeochemical models.

Finally, using observed present-day nutrient climatologies for studies of the past and future evolution of the ocean carbon cycle, where production-remineralization-burial patterns differ from the present ones, may possibly lead to even larger errors than those that we are trying to address here.

We agree that it would be inappropriate to use present-day nutrient climatologies when studying the distant past. For the future though, e.g., year 2100, we would expect that overall, larger errors would be introduced by assuming that nutrient concentrations are always zero. A better solution for carbonate chemistry calculations in models that carry dissolved inorganic nitrogen but not dissolved inorganic phosphorus would be to compute the latter from the former assuming a constant Redfield ratio (as suggested by Referee 3). We will bring up these points in the revised manuscript.

2 Specific comments

In the following, when reference is made to CO2SYS, it should be understood that it is the MATLAB version that is meant (van Heuven et al., 2011), the only version for which that I was able to inspect the source code.

Page 2880, lines 18–19: this is rather cryptic. Does this mean that the errors listed by Lewis and Wallace (1998) for Millero (1995) have been taken into account? If so, it would be helpful to make the text more precise.

Yes, errors listed by Lewis and Wallace (1998) have been taken into account in mocsy. We use the same pressure adjustment coefficients as given in our companion paper (Orr et al., 2014, Table 7). We will emphasize these details in the revised manuscript.

Page 2881, line 6: PO_4^{3-} is ambiguous (is it the species or the total dissolved phosphate that is meant?). I guess it should read “total phosphate”.

In the revised manuscript, we will systematically replace PO_4^{3-} and SiO_2 with total dissolved inorganic phosphorus P_T and total dissolved inorganic silicon Si_T, respectively.

Page 2881, lines 14–16: it was previously said (p. 2880, ll. 18–19), that the pressure-correction coefficients came from Lewis and Wallace (1998). Here we find other references. Are those the same coefficients? If so, the original reference should be cited; if not, how do they differ? Please clarify.

Thanks for signaling the potential for reader confusion between these 2 passages. There are differences between the two, because they do not refer to the same code. In the cited passage on p. 2880, we are discussing the mocsy code; in the subsequent passage on p. 2881 we are referring to its precursor code that was never published. Some of the pressure adjustment coefficients in the precursor code were erroneous which lead to minor differences in computed variables as shown in our companion manuscript. We will make things clearer in the revised manuscript. (Orr et al., 2014).

Page 2881, lines 10–12: The description of the method used to solve the alkalinity-pH equation is insufficient and does not appear to be correct:

1. Maier-Reimer (1993) uses the practical alkalinity approximation and specifies that a Newton iteration is used to solve the alkalinity-pH equation, expressed as a variant of the rational function form (see, e.g., Munhoven (2013) for the different forms of the equation);
2. Aumont and Bopp (2006) describe the PISCES model, which, although it derives from HAMOCC5 (Aumont et al., 2003), claims to use the OCMIP protocol (i.e., Newton + bi-section) for its carbonate chemistry, with practical alkalinity as an approximation to total alkalinity;
3. inspection of the MOCSY code shows that the adopted method is actually not

a Newton method, as the cited literature might suggest, but, in the classification of Munhoven (2013), a fixed-point iterative carbonate alkalinity correction (ICAC) method, combined with a pH = 8 initialization scheme.

To the best of our recollection, when O. Aumont was first developing PISCES he tried using the OCMIP code as is, but to solve the alkalinity-pH equation he switched to a faster iterative scheme coded by E. Maier-Reimer. We looked at the PISCES code back in 2004 and adopted its iterative scheme when developing the precursor to mocsy. The same iterative scheme is used in mocsy 1.0. However that will change with mocsy 2.0, for reasons we describe below. In the revised manuscript, we will do our best to clear up any confusion.

It would be worth noting that among the six different solution algorithms studied by Munhoven (2013), icacfp, which is equivalent to the ICAC algorithm adopted here, was the second-least efficient one, two to three times slower than the best (if we only consider the SW1 and SW2 benchmarks that it passed and disregard the SW3 benchmark that it failed). Unless the cubic initialization scheme of Munhoven (2013) was used, it also failed the RTC1 stress-test in about 40–90% of the cases, and the RTC2 stress-test in 77–100% of the cases (Munhoven, 2013, suppl.). The safe-guarded algorithms presented by Munhoven (2013), on the other hand, passed both stress-tests in 100% of the cases, and were at worst 15% slower than the fastest methods in the SW1, SW2, and SW3 benchmarks.

As further shown by Munhoven (2013), ICAC methods present inherent convergence problems at high $AT \div C_T$ ratios and at low C_T values. These convergence problems are unavoidable and can be alleviated only to a very limited extent. There is no way to efficiently safeguard them, as it is the underlying recurrence that becomes divergent. f

ICAC methods are fine for typical present-day conditions, although comparatively slowly converging. It is, however, risky to rely on them for conditions that deviate from present-day. They are, e.g., not the best choice in 3D models that are intended to be used to assess the effects of alkalinization, a geoengineering technique to mitigate ocean acidification and that may locally lead to extremely high A_T concentrations, or the impact of melt water pulses, which might locally lead to too dilute C_T for ICAC methods.

Many of these aspects and possible alternatives are covered in detail by Munhoven (2013).

Dr. Munhoven's arguments are most convincing. As a result, we have now replaced the mocsy 1.0 iterative scheme (ICAC) with the his new algorithm (SolveSAPHE, `solve_at_general`) and included his cubic initialization scheme. Our first tests yield results that are identical to at least the 6th digit after the decimal in terms of pH, but the the new approach is about 5 times faster than our old scheme. Although mocsy is used by its authors for open-ocean conditions (SW1 and SW2), some users may eventually wish to apply it to more extreme cases (such as SW3), so we welcome SolveSAPHE's other benefits. This improvement will be described in the revised manuscript, which will be accompanied by the release of mocsy 2.0. We thank the Referee for emphasizing the features of SolveSAPHE relative to the iterative scheme used in mocsy. His routines provide a major step forward.

The model description needs to be corrected, completed and possible limitations discussed. It could also be better streamlined: e.g., in the current manuscript, the pressure-correction of the equilibrium constants is mentioned in four different places (p. 2880, ll. 18–19; p. 2881, ll. 14–15; p. 2882, ll. 13–15; p. 2884, ll. 2–5), and still, only insiders are likely understand the implications.

We will correct the problems mentioned in the model description. We will also streamline the text regarding pressure corrections. Note that the 1st passage (p. 2881) is a synopsis in the Introduction and the 2nd passage (in the Methods) does not refer to mocsy but its precursor, which was different.

Page 2881, lines 24–28: There are a few more exceptions than these three. Similarly

to K_S, K_F must also be referenced on the free scale (and that is actually how it is implemented in the code). K_0 is also independent on pH scale.

In the revised manuscript, we will also mention that K_F is on the free scale and that K_0 is independent of the pH scale.

Page 2882, line 17: please notice that, if the calcium content is calculated following Riley and Tongudai (1967), as stated, then the correct Ca-to-chlorinity ratio is 0.02128 and not 0.02127, as reported by Dickson et al. (2007, chap. 5, tab. 2). The 0.02127 value has been around in the literature for a while without an adequate reference: it can at least be traced back to Millero (1982), where the cited original references do not allow an unequivocal attribution. I speculate that it was obtained by averaging the 0.02126 from Culkin and Cox (1966) and the 0.02128 from Riley and Tongudai (1967).

We had noticed these tiny differences previously when comparing the source code of different public packages. We are grateful for this reminder, which will prompt us to mention it in the revised manuscript in the context of our next response (below).

The difference is definitely small, but if confusion can be avoided . . .

This small difference may explain why the discrepancy between mocsy and CO2SYS is slightly larger for Ω_A than for CO_3^{2-} (Fig. 2 of the Discussion paper).

Pages 2884–2885, section 3.1: Where do the diagnosed differences in the species concentrations come from? For identical input, I would expect no differences at all (at least to within an order of magnitude of machine precision or so). The differences are not random, so there must be some intrinsic differences between the ways MOCSY and CO2SYS carry out these calculations.

Yes there are some very small systematic differences between mocsy and CO2SYS. We have already provided a preliminary response to the same question from another Referee (Andrew Dickson). We will report back more fully in the revised manuscript.

The reported discrepancies in the Revelle factor are quite interesting: the largest differences arise where the nutrient contributions to alkalinity are the largest. This points out an important shortcoming in MOCSY's Revelle factor calculation: the implemented analytical buffer factor formulation of Frankignoulle (1994) was derived for the practical alkalinity approximation. CO2SYS, on the other hand, uses a finite difference approach to calculate the Revelle factor, using results derived with the (most) complete alkalinity representation. Despite being based upon a numerical approximation only, the CO2SYS approach might be more reliable, since it is consistent with the adopted A_T approximation.

We concur. We will address this in the revised manuscript.

Pages 2885–2886, section 3.2: this section is not totally accurate and partially contradictory.

1. The 4% increase of the total borate concentration, B_T, does not translate into a 4% increase of borate alkalinity, A_B, as could be easily verified with, e.g., MOCSY. There is some buffering. Indeed, if total alkalinity, A_T, stays constant, together with all the other total concentrations, pH will adjust (decrease), such as to decrease all of the other alkalinity contributions besides A_B. Calculations with SolveSAPHE 1.0.1 (Munhoven, 2013) indicate that that 4% B_T increase leads to A_B increase of close to 3% (using the practical alkalinity approximation for the sake of simplicity).
2. The B_T increase does not affect A_T as stated, nor does it affect C_T (also an input variable). The conclusion that "the C_T must also decline" (p. 2886, l. 12) does therefore not make sense. While it is correct that carbonate alkalinity A_C must decline, as a result of the increased AB, this decline is brought about by a pH decrease, not by a C_T change. The buffer capacity is dependent on the actual levels of all of the total dissolved acid concentrations, and on pH (which of course depends on all the previous). With a different pH, a different buffer

capacity is obtained. It is difficult to predict the evolution of the buffer capacity for this change a priori, as its analytical expression involves a complicated product of factors that present counteracting variations.
3. For the same reasons (i.e., A_T and C_T are input variables), and contrary to what is written at ll. 12–13, A_T minus C_T remains constant. Here, the commonly made approximation $\left[\text{CO}_3^{2-}\right] \simeq A_T$-$C_T$ is terribly misleading. It would be far more ac curate to use $\left[\text{CO}_3^{2-}\right] \simeq A_C$-$C_T$, which is more compatible with what is observed here.

In the revised manuscript, we will be more precise about the increase in A_B that accompanies a 4% increase in B_T. We slipped when we stated that C_T declines. It contradicts our sentence earlier in that same paragraph that says it cannot change since it is an input variable. We will remedy this confusion in the revised manuscript. We will also offer a simple explanation as detailed in the response to another Referee, Andrew Dickson.

Page 2886, line 16 – page 2886, line 2: issues related to the use of constant density are probably not as important as outlined here (see general comments above).

For the revised manuscript, we will double check our calculations. If they hold up, we will be more quantitative and offer a clearer explanation about the potential errors associated with using a different density to convert back to model concentration units (mol m^{-3}) relative to the constant value assumed when converting from mol m^{-3} to mol kg^{-3} before making the calculations.

Pages 2887–2888: how significant are the calculated differences? Are inter-model differences not far larger than this?

Whether or not intermodel differences are larger, we think that a systematic error of $+6$ μatm in $p\text{CO}_2$ should be fixed, especially when the fix costs so little. In regards to the Southern Ocean, the simulated air-sea CO_2 flux can differ between models by only a tenth of a Pg C yr^{-1}, equivalent to less than 1 μatm in the air-sea difference in $p$$CO_2$ (Dufour et al., 2013, Table 1). A systematic shift of -6 μatm would be enough to switch some models from net CO_2 loss to net CO_2 uptake in some regions such as the Southern Ocean, i.e., for natural CO_2.

Page 2888, line 20: "PO_4^{3-}" is ambiguous—should probably read "total phosphate"

In the revised paper, we will use total dissolved inorganic phosphorus P_T and total dissolved inorganic silicon Si_T.

3 Technical corrections

Page 2880, line 17: strange sentence "[. . .] recommended by for best practices [. . .]" – please check.

In the revised manuscript, we will change "by for" to "for".

Page 2897, Fig. 3: the horizontal axes on the top left and on all of the lower panels should be modified to better separate the different curves. Much space is currently wasted.

We designed Fig. 3 to have the same range (max - min) for each variable as used in subsequent figures (Figs. 4, 5, 6, and 7). Because differences shown in Fig. 3 are generally small, we prefer not to zoom in on them excessively to avoid giving the impression that they are large.

4 Code

4.1 Availability and distribution

Two ways to get the code are indicated, together with a link to a dedicated website of the code, where extensive information is available. The common user who does not have git installed can easily go to the github page and download the zip archive.

I have downloaded, compiled and executed the code on a Linux machine with the gfortran compiler. Everything worked out of the box.

Good to know. Thanks.

4.2 Comments on the code

1. Users will appreciate the extensive comments and references in the source code.
2. `make test_mocsy` produces a `libmocsy.a` library, which is nevertheless not used (not even linked into the produced `test_mocsy`).
3. The code is a mixture of single and double precision operations. While this is perfectly valid, it leads, in my personal opinion and in my experience as a teacher and instructor in Fortran programming, to unnecessary numerical artifacts. Why not use a uniform real type throughout, the more since the flexible typing mechanism offered by Fortran 90 and later is already used?
4. In some of the subroutines one reads in the comments that they are "Needed because xyz is a function (cannot accept arrays)". While this was to some extent true in FORTRAN 77 (arguments could well be arrays, but results not), it is definitely not correct in the Fortran 90 and later standards. Those do also allow array-valued functions. In MOCSY, converting the functions from scalar to array-valued might be interesting for performance reasons, since most (if not all) of the concerned function/subroutine pairs only perform straightforward calculations (no iterations) and might thus take advantage of the efficient whole-array operations.
5. The Dickson and Riley K_F is converted from the free to the total pH scale, to be converted back to the free scale right away before being corrected for pressure. The two conversion steps could of course be skipped (and the Perez and Fraga K_F directly converted to the free scale instead).

1. Thanks.
2. We have changed the makefile so that the test program `test_mocsy` links with `libmocsy.a` instead of the object files.
3. The mixture of single and double precision is intentional. It allows calculations to be performed at double precision but input and output to mocsy to be single precision. To save disk space, most model output such as from CMIP5 is stored as single precision. Individual users who wish to change everything to double precision can do so easily because mocsy exploits the flexible typing system of Fortran 90. Changing the type in the module `singledouble.f90` would change the type throughout the code.
4. The mocsy code was recently converted from Fortran 77 to Fortran 90. We will consider the possibility of converting some functions so that arguments are passed as arrays not scalars. And we will remove the comment.
5. We convert K_F from Dickson and Riley (1979) from the free to the total scale so that it can be treated consistently (later in the code), i.e., just as the alternative case when the user chooses to use K_F from Perez and Fraga (1987), which is on the total scale to start with.

References

Dickson, A. G. and Riley, J. P.: The estimation of acid dissociation constants in seawater media from potentiometric titrations with strong base. I. The ionic product of water - K_W, Mar. Chem., 7, 89–99, 1979.

Dufour, C. O., Sommer, J. L., Gehlen, M., Orr, J. C., Molines, J.-M., Simeon, J., and Barnier, B.: Eddy compensation and controls of the enhanced sea-to-air CO2 flux during positive phases of the Southern Annular Mode, Global Biogeochem. Cycles, 27, 950–961, doi:10.1002/gbc.20 090, 2013.

Lewis, E. and Wallace, D. W. R.: Program Developed for CO_2 System Calculations, ORNL/CDIAC-105, Carbon Dioxide Inf. Anal. Cent., Oak Ridge Natl. Lab., Oak Ridge, Tenn., 38 pp., 1998.

Orr, J. C., Gattuso, J.-P., and Epitalon, J.-M.: Comparison of seven packages that compute ocean carbonate chemistry, Biogeosciences Disc., 11, 5327–5397, doi:10.5194/bgd–11–5327–2014, 2014.

Perez, F. F. and Fraga, F.: Association constant of fluoride and hydrogen ions in seawater, Mar. Chem., 21, 161–168, 1987.