Contents

Chapter 4

A data assimilating state-space model for algal growth under controlled conditions within a photobioreactor

4.1 Introduction

Microalgae are tiny organisms..

[BM for ref: At 2 hour intervals, a solenoid valve (SMC Pneumatics Pty. Ltd.) was used to stop aeration for 10 minutes. The linear increase in DO caused by these artefacts were used to calculate net photosynthesis. (reference Tamburic 2015)]

4.2 Methods

4.2.1 Data Model: Photobioreactor setup, experimental design and data collection methods

All data collection methods for this chapter were carried out by Peter Wood as part of a collection of PhD experiments (Peter Wood 2019 UTS PhD).

Microalgal culture *Nannochloropsis oceanica* (Droop) Green (strain CS-179) obtained from the Australian National Algae Culture Collection was cultured in 200 mL conical flasks; maintained in an incubator (Labec Pty Ltd) at 20°C, under an irradiance of 50 μmol m^{-2} s^{-1} of cool-white fluorescent light at a 12 hour light/12 hour dark cycle. Stock cultures were grown in f/2 saltwater medium [4] and diluted 5 days prior to the start of experiments to ensure that *N. oceanica* was in the exponential growth phase and not nutrient deprived. f/2 was sparged prior to stock culture dilutions to maximise carbon and oxygen content.

N. oceanica was cultured in four, 500 mL environmental photobioreactors (ePBRs, Phenometrics Inc) with a 10% v/v inoculation of stock culture. Top-side illumination over a path length of 25 cm was provided by a cool-white light LED, whilst temperature was maintained at 27°C using a Peltier heater-cooler connected to a water jacket. In-built thermocouples, calibrated against external temperature sensors attached to the Firesting module (TeX4; PyroScience GmbH), measured every 5 minutes were used to control the Peltier heater-cooler jacket through a feedback loop to an accuracy of ± 0.2°C. pH was also measured in 5 min intervals by in-built pH electrodes (Van London Inc); controlled by periodic CO2 (5%) injections using valves in the ePBRs. pH was 3-point calibrated using pH buffer solutions at pH 4.00 ± 0.02, pH 7.00 ± 0.02 and pH 10.00 ± 0.02. PBR mixing was controlled by magnetic stirring bars at 110 rpm. All four ePBRs were aerated with filtered/humidified air through a 1.2 mm needle valve (Terumo Co).

A period of 2 days was allowed for *N. occulata* to acclimate to the ePBRs at an irradiance of 500 μmol m^{-2} s^{-1} and a temperature of 27°C. Following this acclimation period, the PBR was set to the experimental condition of 2,000 μmol photons m^{-2} s^{-1} for another 2 days and a 12 hour light/12 hour dark cycle with a temperature of 27°C. ePBRs were maintained at an optical density (OD) of 0.4 using manual dilutions, creating a semi-batch culturing system. Dilutions occurred once per day (one hour before the light cycle), using aerated f/2 media. The experiment was conducted over a period of 4 days, samples were extracted post and prior dilution, as well as half way through the light cycle. 50 mL was extracted to examine total alkalinity and dissolved inorganic carbon. Dissolved oxygen (DO) was measured using a 3 mm robust optical probe (OXROB10-OI; PyroScience GmbH) attached to a FireStingO2 logger (PyroScience GmbH). DO measurements were taken every 60 seconds and temperature-corrected using a temperature extension module (TeX4; PyroScience GmbH). DO was two-point calibrated using air-saturated seawater (100% saturation) and sodium sulfate-saturated water (0% saturation). At 2 hour intervals, a solenoid valve (SMC Pneumatics Pty. Ltd.) was used to stop aeration for 10 minutes to allow for observations of net photosynthesis. Alkalinity was measured twice a day using TA titration; 0.1 M hydrochloric acid on 30 mL of *N. oceanica* media using an auto-titrator (800 Dosino; Metrohm AG).

[Chris: DIC measurement collection description]

[BM: confirm that reference with PW]

4.2.2 Data model: Data treatment, distributions and measurement error

Valve, temperature, light (normalised to 0/1) and dilution rates were used to force the model. Dissolved oxygen, pH, dissolved inorganic carbon and total alkalinity observations for 4 days post acclimation were assimilated. While pH observations were calibrated and corrected, it was visible that O_2 observations were not completely calibrated and experienced some sensor drift during the experiment. An offset term ($offset_{O_2}$) was added to the O_2 ode to account for this. The offset was assigned a normally distributed prior distribution with mean 0 and standard deviation 2.

The data model assigned log normally distributed observation errors for each instrument; $\mathrm{O}_{2_{obs}} \sim \mathrm{Log}\mathcal{N}(\log(\mathrm{O}_2),\, \sigma_{O_2})$, $\mathrm{pH}_{obs} \sim \mathrm{Log}\mathcal{N}(\log(\mathrm{pH}),\, \sigma_{pH})$, $\mathrm{DIC}_{obs} \sim \mathrm{Log}\mathcal{N}(\log(\mathrm{DIC}),\, \sigma_{DIC})$, $\mathrm{TA}_{obs} \sim \mathrm{Log}\mathcal{N}(\log(\mathrm{TA}),\, \sigma_{DIC})$, where the standard deviations (σ_{O_2}, σ_{pH}, σ_{DIC}) were unknown parameters to be estimated as part of the assimilating model. Dissolved inorganic carbon and total alkalinity measurements were obtained from the same instrument thus the error is shared between these states. Initial observation error priors started at $\sigma_{O_2} \sim \mathrm{Log}\mathcal{N}(\log(0.1),\, 0.5)$ and then were adjusted during the PMMH tuning phase.

[Chris/SW: should I talk about the thinning out of O_2 and pH obs?]

4.2.3 Process model: Carbon chemistry

To calculate the carbon chemistry of the photo-bioreactor, we would ideally use CO2SYS [5] to calculate HCO_3^-, CO_2, CO_3 and pH. CO2SYS is a program developed for CO_2 system calculations (CO2SYS) that calculates and returns a detailed state of the carbonate system of oceanographic water samples in seawater and freshwater [5]. It uses two of the four measurable carbonate system parameters (total alkalinity, total inorganic CO_2, pH, and either fugacity fCO_2 or partial pressure of CO_2) to calculate the other two parameters at a set of input conditions (temperature and pressure).

To incorporate CO2SYS into LiBbi for solving carbon chemistry on the timescale of the microalgae model, we explicitly define 2 iterations of the Newton-Raphson method for finding approximations to roots of real valued functions. The Newton-Raphson method is an iterative process considering a function, its derivative and an initial starting value. Vital to the convergence of the Newton-Raphson method is a good starting value. To provide a good starting value, we randomly sample from a range of CO2SYS input parameters (temperature = 20-30, salinity = 30-40, DIC = 200-2500, and alkalinity = 1500-3000) and fit an approximating equation to pH as a function of DIC, S and T (alk?). This gives us a close initial starting value for the Newton-Raphson method.

Converges in 2-3 iterations

Choice of H2CO3 and HCO3- dissociation constants K1 and K2 was Mehrbach (refit BY DICKSON AND MILLERO) [BM: After the iterative approach is finalised, the K1 and K2 constants are adjusted based on measurements taken during the experiment, K1*1.23 and K2*0.53 measured during experiment] temperature: 2-35, salinity: 20-40, Seawater scale, Artificial seawater.

The CO2SYS Matlab version [7] was used to produce values of CO_2 and HCO_3^- across DIC range 200-2500. Approximating equations were fit

Total inorganic CO_2 (TCO_2) is the sum of the dissolved CO_2, the carbonate (CO_{3-2}), and the bicarbonate (HCO_3^-).

4.2.4 Process model: Gas transfer equilibrium concentrations for O_2 and CO_2

The equilibrium concentration for CO_2 solubility in water CO_{2H} (μmol/L) is calculated using Henry's law,

$$CO_{2H} = K0_{CO2} * fCO2 * 1.0220 * 1e6 \tag{4.1}$$

where fCO2 (atm) is the fugacity or approximately the partial pressure of CO_2, 1.0220 is the density of seawater (kg/L) at salinity 34 ppt and temperature 27°C [6] [2]. $K0_{CO2}$ (mol/kg$_{soln}$/atm) is the solubility of gas in seawater [BM: ask Chris: solubility of gas? is this right] and is calculated from the fitted van't Hoff equation and the logarithmic Setchenow salinity dependence [8],

$$\begin{aligned} K0_{CO2} = exp(&-60.2409 + 93.4517(100/T_K) + 23.3585 * ln(T_K/100) + \\ &S(0.023517 - 0.023656(T_K/100) + 0.0047036(T_K/100)^2)) \end{aligned} \tag{4.2}$$

where T_K is the temperature (K) and S is salinity (ppt).

Similarly the equilibrium concentration for O_2 solubility in water O_{2H} is calculated using Henry's law,

$$O_{2H} = K0_{O2} * fO2 * 1.0220 * 1e - 6 \tag{4.3}$$

where fO2 (atm) is the fugacity or approximately the partial pressure of O_2, 1.0220 is the density of seawater (kg/L) at salinity 34 ppt and temperature 27°C [6] [2], and $K0_{O2}$ (mol/kg$_{soln}$/atm) is the solubility of oxygen in seawater with an adjusted salinity dependence [1],

$$K0_{O2} = (exp(-1282.8704 + 36619.96/T_K + 223.1396 * log(T_K) - 0.354707 * T_K$$
$$+S * (5.957e - 3 - 3.7353/T_K) + 3.68e - 6 * S^2))/0.2094e - 6 \tag{4.4}$$

where T_K is the temperature (K) and S is salinity (ppt).

The equilibrium concentrations for O_2 and CO_2 are modelled together with the gas turning on and off during the experiment, as

$$kLA_{O2}\xi(O_{2H} - O_2) \tag{4.5}$$

$$0.893kLA_{O2}\xi(CO_{2H} - CO_2) \tag{4.6}$$

where ξ is the gas state (1= on, 0= off), and kLA_{O2} is the mass transfer coefficient for air (d^{-1}), and 0.893 is the ratio between measured O_2 and CO_2 mass transfer constants [3].

4.2.5 Process model: Photosynthesis and respiration

Net photosynthesis

$$dDIC/dt = -P_1 * I * mm + R_1 \tag{4.7}$$

$$dO_2/dt = \frac{P_1 * I * mm - R_1}{R_Q} \tag{4.8}$$

Photosynthesis (P_1) and respiration (R_1) are both modelled as random walks, by taking P and R, previously constant parameters, and replacing them by $P_1(t)$ and $R_1(t)$. Here, we take $P_1(t)$ and $R_1(t)$ to be such that

$$P_1(t + \Delta t) = P_1(t) + r_P$$

$$R_1(t + \Delta t) = R_1(t) + r_R$$

where $r_P \sim N(0, \sigma_{r_P})$, $r_R \sim N(0, \sigma_{r_R})$, and Δt is the length of discrete time-step. For the purpose of the Bayesian analysis here, σ_{r_P} and σ_{r_R} are treated as a parameter to be inferred.

R_Q is the respiratory quotient, the ratio of CO_2 produced and O_2 consumed by a cell.

PAC is Photosynthetically Active Carbon, this is the type of carbon that the microalgae use for photosynthesis. This can be CO_2, HCO_3^-, or a combination of both, eg $PAC = CO_2 + HCO_3^-$ if the microalgae are using both carbon dioxide and bicarbonate for photosynthesis.

$$PAC = HCO_3^- \tag{4.9}$$

$$mm = \frac{PAC}{K_m + PAC} \tag{4.10}$$

4.2.6 Process model: Ordinary differential equations

Ode's:

$$\begin{aligned}
&\text{Rate} && \text{flux into cells} && \text{gas transfer} && \text{dilution} \\
\frac{\partial DIC}{\partial t} &= && -(P-R) && +\hat{Q}^{air} kLa^{air}_{CO_2}(CO_2^{air} - CO_2) && +\frac{Q^M}{V}(DIC^M - DIC) \\
& && && +\hat{Q}^{co2} kLa^{co2}_{CO_2}(CO_2^{co2} - CO_2) && \qquad (4.11) \\
\frac{\partial O_2}{\partial t} &= && \frac{1}{R_Q}(P-R) && +\hat{Q}^{air} kLa^{air}_{O_2}(O_2^{air} - O_2) && +\frac{Q^M}{V}(O_2^M - O_2) \\
& && && +\hat{Q}^{co2} kLa^{co2}_{O_2}(O_2^{co2} - O_2) && \qquad (4.12) \\
\frac{\partial TA}{\partial t} &= && R_R(P) && && +\frac{Q^M}{V}(TA^M - TA) \\
& && && && \qquad (4.13)
\end{aligned}$$

	Symbol	Description	Prior / Value	Unit
e variable initial conditions	DIC^0	Dissolved inorganic carbon		μM/L
	O_2^0	Oxygen		μM/L
	TA^0	Total alkalinity		μM/L

	P^0	Rate of photosynthesis		μM/L/day
	R^0	Rate of respiration		μM/L/day
	pH^0	-		$\log_{10}$(-mol/L H+)
	${CO_2}^0$	Carbon dioxide		μM/L
	${HCO_3^-}^0$	Bicarbonate		μM/L
	CO_3^{2-0}	Carbonate		μM/L
Gas transfer terms	$\hat{Q}^{air}$	indicator for flow in air line	0 or 1	-
	$x^{air}_{CO_2}$	mole fraction of CO_2 atmosphere	400	ppm
	CO_{2H}	Equilibrium CO_2 concentration	Eq. 4.1	μM/L
	CO_2^{air}	sat CO_2 conc with atmosphere	$x^{air}_{CO_2} CO_{2H}$	
	$kLa^{air}_{CO_2}$	Mass transfer coefficient for CO_2	$0.893 kLa^{air}_{O_2}$	day^{-1}
	$x^{air}_{O_2}$	mole fraction of O_2 atmosphere	0.2094	atm
	O_{2H}	Equilibrium O_2 concentration	Eq. 4.3	μM/L
	O_2^{air}	sat O_2 conc with atmosphere	$x^{air}_{O_2} O_{2H}$	
	τ	half-life of $kLa^{air}_{O_2}$	range(2-20)	min^{-1}
	$kLa^{air}_{O_2}$	Mass transfer coefficient for O_2	$\ln(2) * 24 * 60/\tau$	day^{-1}
Dilution terms	Q^M	rate		ml/day
	V	Volume of the reactor	500	ml
	DIC^M	Media dissolved inorganic carbon	1724.20	μM/L
	O_2^M	Media oxygen concentration	226.65	μM/L
	TA^M	Media total alkalinity	1797.90	μM/L
Other dilution terms	$\hat{Q}^{CO2}$	indicator for dilution	0 or 1	-
	$x^{CO2}_{O_2}$	mole fraction of	0	-
	O_2^{CO2}	sat CO_2 conc with CO_2	$x^{CO2}_{O_2} O_{2H}$	
	$kLa^{CO2}_{O_2}$	mass transfer coefficient		day^{-1}
	O_2^{CO2}			
	$x^{CO2}_{CO_2}$	mole fraction of	1	ppm

	CO_2^{CO2}	sat CO_2 conc with CO_2	$x_{CO_2}^{CO2} CO_{2H}$	
	$kLa_{CO_2}^{CO2}$	mass transfer coefficient	$0.893 kLa_{O_2}^{CO2}$	day^{-1}
	CO_2^{CO2}			

Symbol	Variable	Units
DIC	Dissolved inorganic carbon concentration	μmol/L
O_2	Oxygen	μmol/L
pH	-	$\log_{10}$(-mol/L H+)
CO_2	Carbon dioxide	μmol/L
HCO_3^-	Bicarbonate	μmol/L
CO_3^{2-}	Carbonate	μmol/L
PAC	Photosynthetically active carbon	μmols/L
mm	-	-
kLA_{O2}	Mass transfer coefficient for O_2	d^{-1}
CO_{2H}	Equilibrium CO_2 concentration	μmols/L
$K0_{O2}$	Solubility of gas	mol/kg_{soln}/atm
$K0_{CO2}$	Solubility of gas	mol/kg_{soln}/atm
TA	Total alkalinity	μmols/L
S	Salinity	ppt
fCO2	Fugacity/CO_2 partial pressure	atm
fO2	Fugacity/O_2 partial pressure	atm
K_m	Carbon restriction	μmols/L
P	Photosynthesis rate	μmols/L/day
R	Respiration rate	μmols/L/day
R_R	Redfield ratio	-
R_Q	Respiratory quotient	-

Table 4.2 : Table of variables and parameters.

Symbol	Variable	Units
I	Light Intensity	normalised to 0-1
T	Temperature	$\circ$ C
T_K	Temperature	K
ξ	gasflow	on/off (1,0)

Table 4.3 : Table of Forcings

4.2.7 Parameter Model: Priors

Decide whether the parameters vary in time or not.

Parameter	Prior	Proposal
S	34	*
fCO2	397e-6	*
fO2	0.21	*
kLA_{O2}	LogNormal(log(200.0), 0.5)	LogNormal(log(kLA_{O2}), 0.5$prop_{std}$)
K_m	LogNormal(log(200.0), 0.8)	LogNormal(log(K_m), 0.8$prop_{std}$)
R_R	Uniform(0.0001, 0.2)	TrunNormal(R_R, 0.2$prop_{std}$, 0.0001, 0.2)
R_Q	Uniform(0.66, 1)	TrunNormal(R_Q, 0.2$prop_{std}$, 0.66, 1.0)
σ_P	Normal(0.05, 0.01)	Normal(σ_P, 0.01$prop_{std}$)
σ_R	Normal(0.01, 0.001)	Normal(σ_R, 0.001$prop_{std}$)

Table 4.4 : Table of Parameters, their priors and proposal distributions. * indicates the parameter was held fixed. ($prop_{std}$ =0.1)

4.3 Results

4.3.1 Carbon chemistry iterative solution

Total Sulfur

$$
\begin{aligned}
TS &= \frac{0.14}{96.062} * \frac{S}{1.8065} \\
IS &= 19.924 * \frac{S}{(1000.0 - 1.005 * S)} \\
KS_{int} &= -\frac{4276.1}{T_K} + 141.328 - 23.093 * log(T_K) + (-\frac{13856.0}{T_K} + 324.57 \\
&\quad - 47.986 * log(T_K)) * \sqrt{IS} + (\frac{35474}{T_K} - 771.54 + 114.723 * log(T_K)) * IS \\
&\quad - \frac{2698}{T_K} * IS^{1.5} + \frac{1776}{T_K} * IS^2 \\
KS &= exp(KS_{int}) * (1 - 0.001005 * S)
\end{aligned}
$$

Fluorine

$$
\begin{aligned}
TF &= 0.000067 * S/18.9984/1.80655 \\
KF &= exp(-(-\frac{874.0}{T_K} - 0.111 * \sqrt{S} + 9.68)) \\
SWS_{2_T} &= \frac{(1 + \frac{TS}{KS})}{(1 + \frac{TS}{KS} + \frac{TF}{KF})} \\
Free_{2_T} &= 1 + \frac{TS}{KS}
\end{aligned}
$$

H2O dissoc

$$KW = exp(148.9802 - \frac{13847.26}{T_K} - 23.6521 * log(T_K) + (\frac{118.67}{T_K} - 5.977 + 1.0495 * log(T_K)) * \sqrt{S} - 0.01615 * S)$$

Boron

$$KB = exp((-8966.90 - 2890.53 * \sqrt{S} - 77.942 * S + 1.728 * S * \sqrt{S} - 0.0996 * S^2)/T_K + 148.0248 + 137.1942 * \sqrt{S} + 1.62142 * S - (24.4344 + 25.085 * \sqrt{S} + 0.2474 * S) * log(T_K) + 0.053105 * \sqrt{S} * T_K)$$

$$TB = 0.0004326 * \frac{S}{35}$$

Carbon eq constants

$$K1 = 10^{(-(\frac{3633.86}{T_K} - 61.2172 + 9.6777 * log(T_K) - 0.011555 * S + 0.0001152 * S^2))} * 1.23$$

$$K2 = 10^{(-(\frac{471.8}{T_K} + 25.9290 - 3.16967 * log(T_K) - 0.01781 * S + 0.0001122 * S^2))} * 0.53$$

1.23 and 0.53 were experiment specific and measured

Initial guess at the pH

$$pH_{init} = 12.26 - 0.0030605 * DIC - 0.043752 * T - 0.013625 * S + 0.00011315 * TA + 1.3463e - 5 * DIC * T + 5.2215e - 7 * DIC * TA$$

Iterations:

For 1 to n iterations:

$$
\begin{aligned}
h &= 10^{-pH_{init}} \\
h_{free} &= \frac{h}{Free_{2_T}} \\
f0 &= (DIC * 1e-6 * \frac{K1 * h + 2 * K1 * K2}{h * h + K1 * h + K1 * K2} \\
&\quad - h_{free} + \frac{KW}{h} - Alk * 1e-6 + \frac{TB}{1 + \frac{h}{KB}}) * 1e6 \\
df0 &= (DIC * 1e-6 * \frac{K1 + 2 * K1 * K2}{h^2 + K1 * h + K1 * K2} \\
&\quad - DIC * 1e-6 * \frac{(K1 * h + 2 * K1 * K2)}{(h^2 + K1 * h + K1 * K2)^2} * (2 * h + K1) \\
&\quad - TB * \frac{1}{(1 + \frac{h}{KB})^2} / KB \\
&\quad - \frac{KW}{h^2} - \frac{1}{Free_{2_T}}) * 1e6 * (-log(10) * 10^{-pH}) \\
pH &= pH - \frac{f0}{df0} \\
H &= 10^{-pH} \\
denom &= H^2 + K1 * H + K1 * K2 \\
CO2 &= \frac{DIC * H^2}{denom} \\
HCO3 &= \frac{DIC * H * K1}{denom} \\
CO3 &= \frac{DIC * K1 * K2}{denom}
\end{aligned}
$$

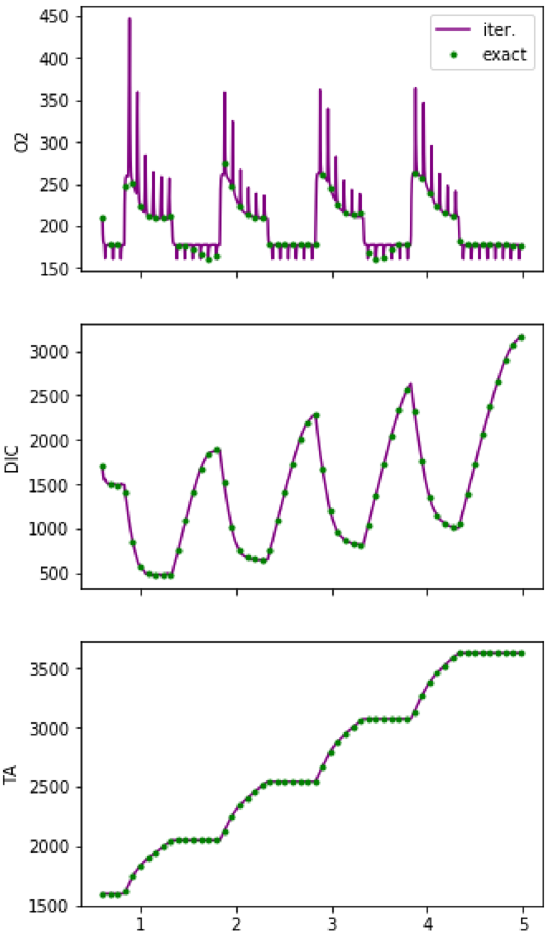

Figure 4.1 : Iterative vs exact solution for state variables O_2, DIC, and TA.

Variable	**Iter. 1**	**Iter. 2**	**Iter. 3**	**Iter. 4**	**Iter. 5**
O_2	0.308389964	0.016044284	4.18E-05	6.89E-05	7.59E-05
DIC	16.78775711	0.958511825	0.005229318	0.002305054	0.002333411
TA	2.607767674	0.160897272	0.000688102	0.001257725	0.001218981
pH	0.036092734	0.002355758	1.41E-05	6.93E-06	6.93E-06
$CO2$	2.109401968	0.145719349	0.001222812	0.000866728	0.000866727
$HCO3$	19.81869214	1.21021115	0.008016765	0.001025002	0.001025139
$CO3$	20.89660704	1.307061652	0.00867642	0.001102278	0.001102434

Table 4.5 : RMSE for 5 iterations of the Newton-raphson carbon chemistry iterative solution.

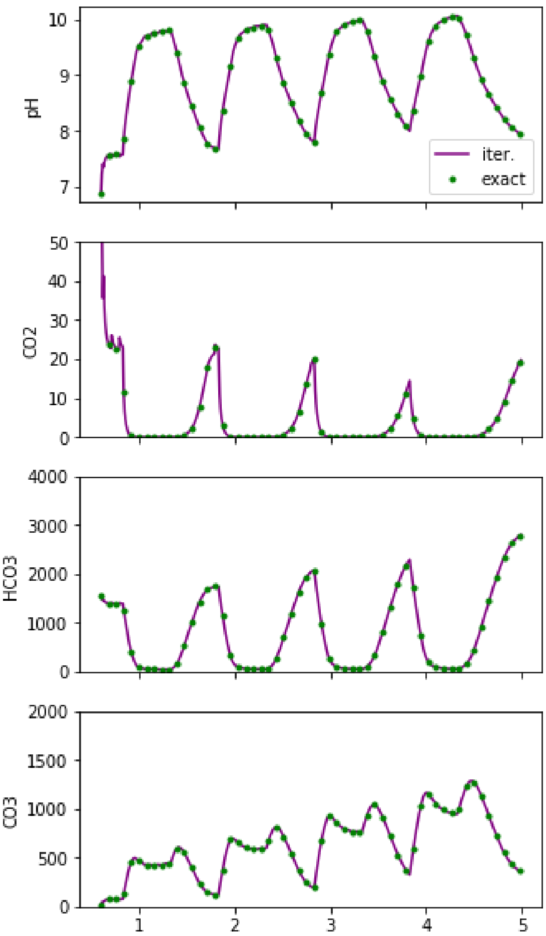

Figure 4.2 : Iterative vs exact solution for carbon chemistry CO_2, HCO_3, CO_3, and pH.

4.3.2 Posteriors

[BM: Talk about the tuning of the observation error priors σ_{O_2} etc]

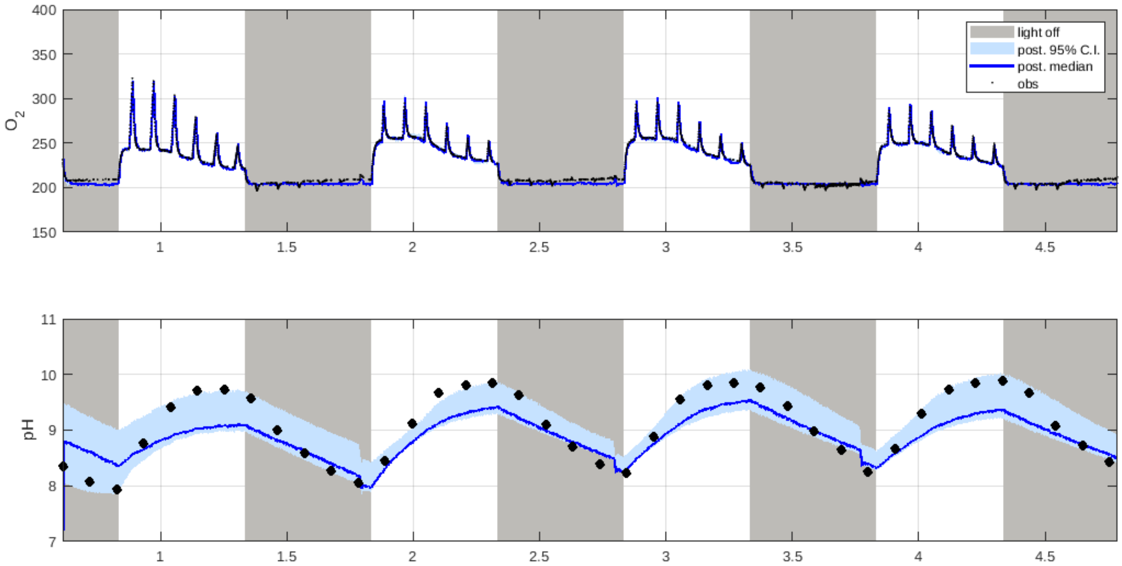

Figure 4.3 : Posteriors for O_2 and pH.

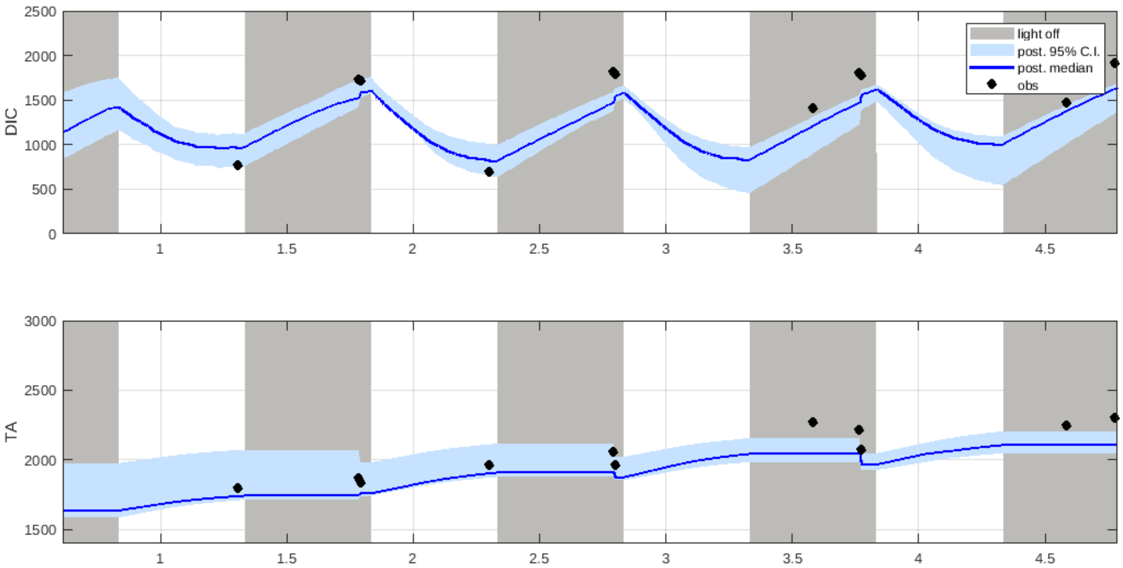

Figure 4.4 : Posteriors for DIC and TA.

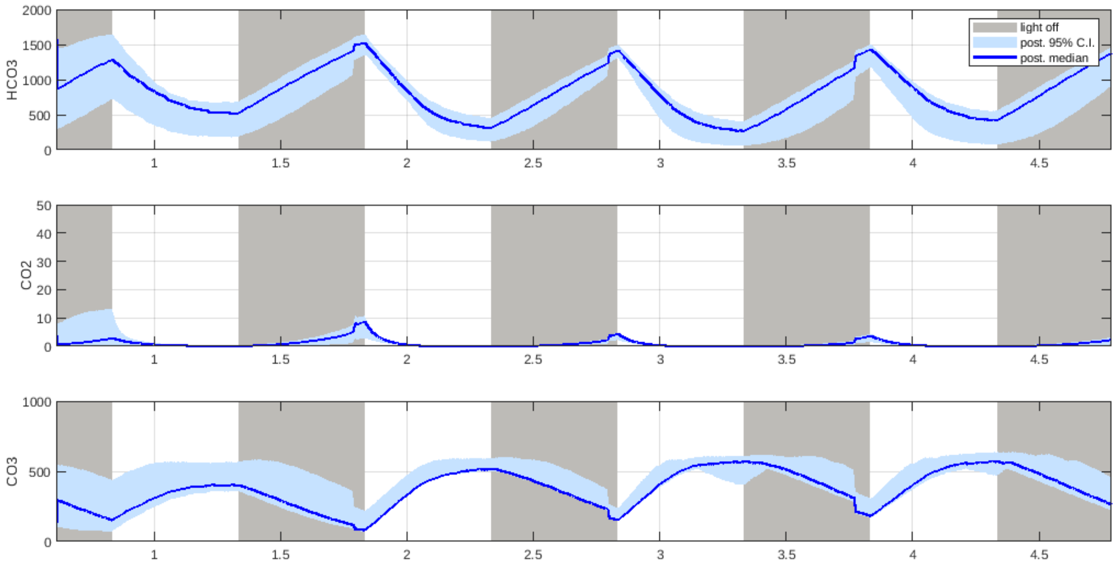

Figure 4.5 : Posteriors for carbon chem.

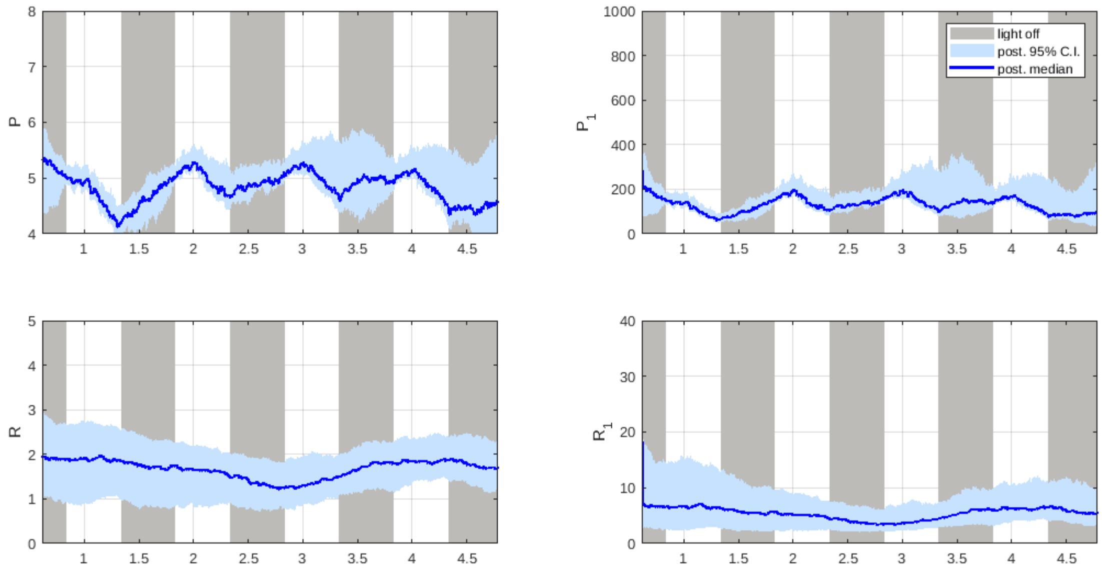

Figure 4.6 : Posteriors for photosynthesis and respiration.

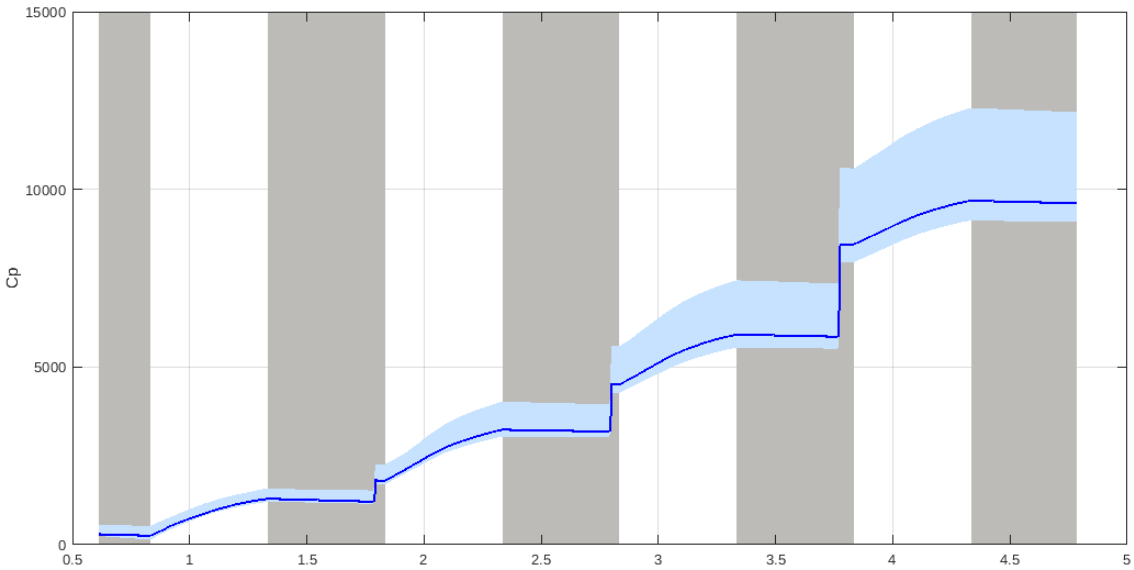

Figure 4.7 : Posterior for C_p.

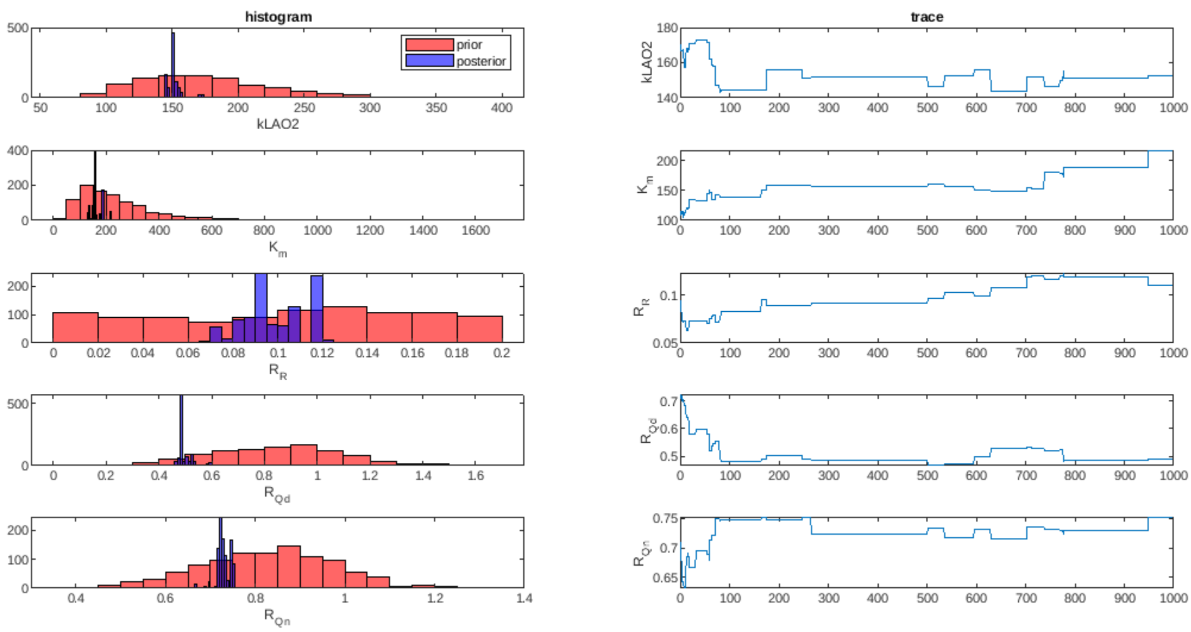

Figure 4.8 : Priors, posteriors and traces for model parameters.

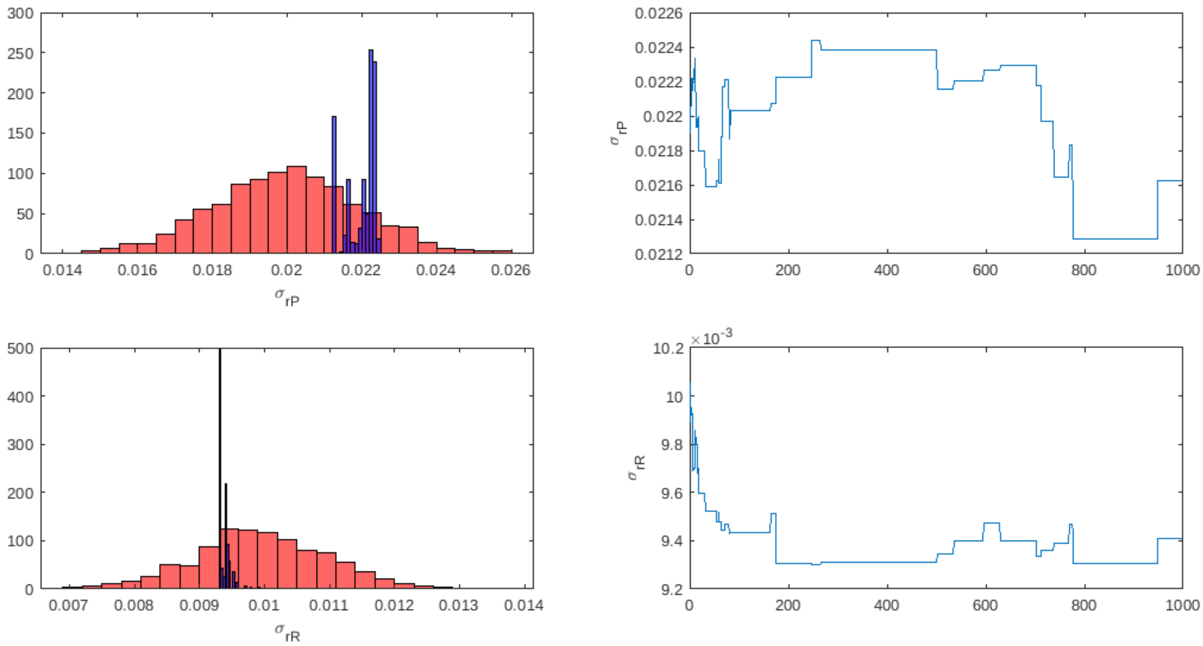

Figure 4.9 : Priors, posteriors and traces for model parameters.

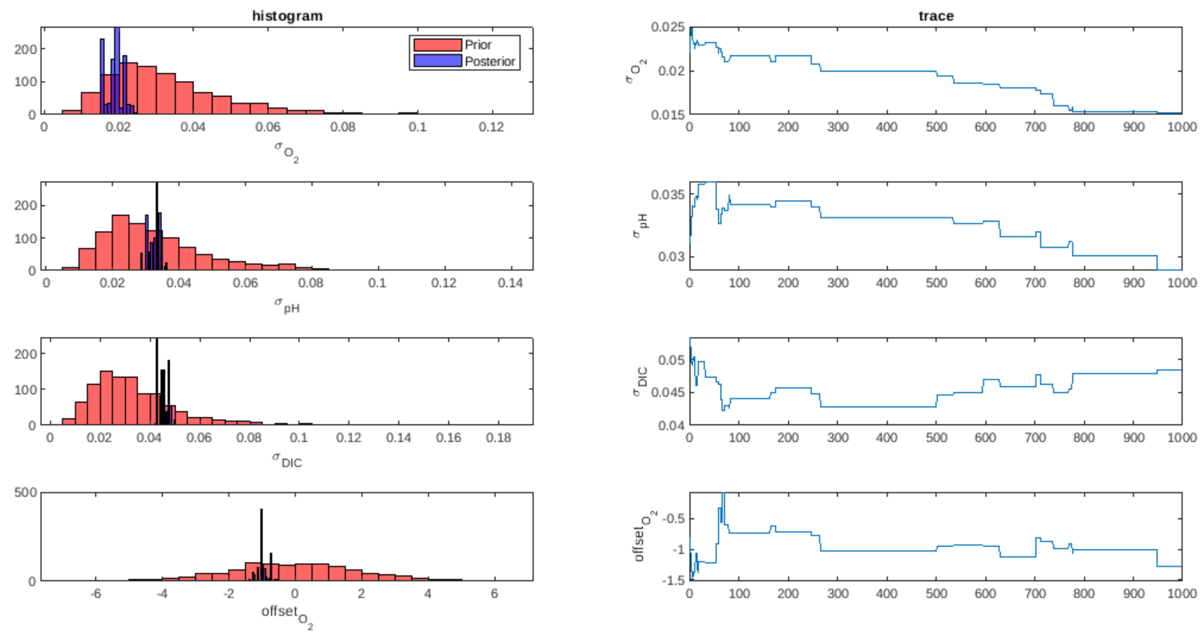

Figure 4.10 : Priors, posteriors and traces for obs. error parameters.

Appendix A

LiBbi model code

LiBbi model file: micro_iterative.bi

```
model micro_iterative {

const FO2       = 0.2094
const FCO2      = 397e-6
const S         = 34.0
const V         = 500.0                // volume of the reactor
const DIC_M     = 1724.20              // calculated with CO2SYS[DIC_M = 1724.20, Alk = 1797.90, T = 27, S = 34]
const O_2_M     = 226.65
const alk_M     = 1797.90
const tau       = 6.0
const kLAO2_m   = log(2.0)*24.0*60.0/tau

param kLAO2
param Km
param RR
param RQ_d
param RQ_n
param sigma_O_2
param sigma_pH
param sigma_DIC
```

```
param offset_O_2

input I                    // light intensity
input T                    // temperature (C)
input gas                  // gas on/off
input dil                  // dilution rate

state DIC // state variables
state O_2
state pH
state Cp
state mich_ment
state O2H_pr
state CO2H_pr
state R
state R1
state P
state P1
state alk
state CO2
state HCO3
state CO3
state O_2H
state CO2H
state h_3
state h_free_3

noise r_R
noise r_P
```

```
/* random walk parameter */
param sigma_r_R
param sigma_r_P

obs O2_obs
obs pH_obs
obs DIC_obs
obs alk_obs

sub parameter {/* prior distribution over parameters */
Km        ~ log_normal(log(100.0), 0.5)
kLAO2     ~ log_normal(log(kLAO2_m), 0.3)
RR        ~ uniform(0.0001, 0.2)
RQ_d      ~ uniform(0.66, 1.0)
RQ_n      ~ uniform(0.66, 1.0)

sigma_O_2 ~ log_normal(log(0.03), 0.5)
sigma_pH  ~ log_normal(log(0.03), 0.5)
sigma_DIC ~ log_normal(log(0.03), 0.5)

offset_O_2 ~ normal(0, 2.0)

sigma_r_R        ~ normal(0.01, 0.001)
sigma_r_P        ~ normal(0.05, 0.01)
}

const prop_std = 0.1;
sub proposal_parameter {
```

```
Km       ~ log_normal(log(Km), 0.5*prop_std)
kLAO2    ~ log_normal(log(kLAO2), 0.3*prop_std)
RR       ~ truncated_normal(RR, 0.2*prop_std, lower = 0.0001,
   upper = 0.2)
RQ_d     ~ truncated_normal(RQ_d, 0.2*prop_std, lower = 0.66,
   upper = 1.0)
RQ_n     ~ truncated_normal(RQ_n, 0.2*prop_std, lower = 0.66,
   upper = 1.0)


sigma_O_2       ~ log_normal(log(sigma_O_2), 0.5*prop_std)
sigma_pH        ~ log_normal(log(sigma_pH), 0.5*prop_std)
sigma_DIC       ~ log_normal(log(sigma_DIC), 0.5*prop_std)

offset_O_2      ~ normal(offset_O_2, 2.0*prop_std)

sigma_r_R       ~ normal(sigma_r_R, 0.001*prop_std)
sigma_r_P       ~ normal(sigma_r_P, 0.01*prop_std)
}

sub initial {/* prior distribution over initial conditions,
   given parameters */
// specify the initial condition model
R        ~ normal(log(20.0), 0.4)
R1       ~ log_normal(log(20.0), 0.4)
P        ~ normal(log(200.0), 0.4)
P1       ~ log_normal(log(200.0), 0.4)

Cp       ~ log_normal(log(300.0), 0.2)
```

```
alk      ~ log_normal(log(1750.0), 0.1)
DIC      ~ log_normal(log(1300.0), 0.2)
O_2      ~ log_normal(log(225.0), 0.2)
pH       ~ log_normal(log(8.5), 0.2)
CO2      ~ log_normal(log(3.0), 0.4)
HCO3     ~ log_normal(log(1000.0), 0.3)
CO3      ~ log_normal(log(300.0), 0.4)
O_2H     ~ log_normal(log(200.0), 0.2)
CO2H     ~ log_normal(log(10.0), 0.2)
}


//sub transition(delta = 0.0023) { // obs are in days ie
   delta=1.0 for daily solving. delta=0.00069 for solving
   every minute, 0.0014 for every 2 mins, 0.0021 for 3 mins,
   0.0028 for 4mins, delta=0.000011574 for solving every
   second
sub transition(delta = 0.0021) {

/* processes */

inline TK     = T + 273.15              // temp in kelvin
inline K0_CO2 = exp(-60.2409 + 93.4517*(100.0/TK) + 23.3585*
   log(TK/100.0)+ S*(0.023517 - 0.023656*(TK/100) +
   0.0047036*(TK/100.0)*(TK/100.0)))
CO2H          <- K0_CO2*FCO2*1.0220*1e6

inline K0_O2  =  (exp(-1282.8704 + 36619.96/TK + 223.1396*log
   (TK) -0.354707*TK + S*(5.957e-3 -3.7353/TK) + 3.68e-6*S*S)
```

```
)/(0.2094e-06)
O_2H          <- K0_O2*FO2*1.0220*1e-6

inline PAC    = HCO3            //PAC=photosynthetically active carbon. if the phyto are just using CO2 to photosynthesise then PAC=CO2
inline mm     = PAC/(Km + PAC)

// CO2SYS iterative solution
// set up all the constants

inline logTK  = log(TK)
inline S2     = S*S
inline sqrtS  = sqrt(S)

// total sulphur

inline TS     = (0.14/96.062)*(S/1.80655)
inline IS     = 19.924*S/(1000.0 - 1.005*S)

inline KS_int = -4276.1/TK + 141.328 - 23.093*logTK + (-13856.0/TK + 324.57  -  47.986*logTK)*sqrt(IS) + (35474.0/TK - 771.54  + 114.723*logTK)*IS - 2698.0/TK*IS**1.5 + 1776.0/TK*IS**2
inline KS     = exp(KS_int)*(1 - 0.001005*S)

// Fluorine

inline TF       = 0.000067*S/18.9984/1.80655
```

```
inline KF       = exp(-(-874.0/TK - 0.111*sqrtS + 9.68))
inline SWS_2_T  = (1.0 + TS/KS)/(1.0 + TS/KS + TF/KF)
inline Free_2_T = 1.0 + TS/KS

// H2O dissoc

inline KW = exp(148.9802 - 13847.26/TK  - 23.6521*logTK + (118.67/TK - 5.977 + 1.0495*logTK)*sqrtS - 0.01615*S)

// Boron

inline KB = exp((-8966.90 - 2890.53*sqrtS - 77.942*S + 1.728*S*sqrtS - 0.0996*S2)/TK + 148.0248 + 137.1942*sqrtS + 1.62142*S - (24.4344 + 25.085*sqrtS + 0.2474*S)*logTK + 0.053105*sqrtS*TK)
inline TB = 0.0004326*S/35.0

// Carbon eq constants

inline K1 = 10**(-(3633.86/TK - 61.2172 + 9.6777 *logTK - 0.011555*S + 0.0001152*S**2))*1.23    //1.23 experiment specific and measured
inline K2 = 10**(-( 471.8/TK + 25.9290 - 3.16967*logTK - 0.01781*S + 0.0001122*S**2))*0.53      //0.53 experiment specific and measured

// end all the constants

// intial guess at the pH (use the approximating equation)
```

```

inline pH_init = 12.26 -0.0030605*DIC -0.043752*T -0.013625*S + 0.00011315*alk + 1.3463e-05*DIC*T + 5.2215e-07*DIC*alk

// iteration 1

inline h_1      = 10.0**(-pH_init)
inline h_free_1 = h_1/Free_2_T
inline f0_1     = (DIC*1e-6*(K1*h_1 + 2.0*K1*K2)/(h_1*h_1 + K1*h_1 + K1*K2) - h_free_1 + KW/h_1 - alk*1e-6 + TB/(1.0 + h_1/KB))*1e6
inline df0_1    = (DIC*1e-6*(K1 + 2.0*K1*K2)/(h_1**2.0 + K1*h_1 + K1*K2) - DIC*1e-6*(K1*h_1 + 2.0*K1*K2)/(h_1**2.0 + K1*h_1 + K1*K2)**2.0*(2.0*h_1 + K1) - TB*1.0/(1.0 + h_1/KB)**2.0/KB - KW/h_1**2.0 - 1.0/Free_2_T)*1e6*(-log(10.0)*10.0**(-pH_init))
inline pH_1     = pH_init - f0_1/df0_1

// iteration 2

inline h_2      = 10.0**(-pH_1)
inline h_free_2 = h_2/Free_2_T
inline f0_2     = (DIC*1e-6*(K1*h_2 + 2.0*K1*K2)/(h_2*h_2 + K1*h_2 + K1*K2) - h_free_2 + KW/h_2 - alk*1e-6 + TB/(1.0 + h_2/KB))*1e6
inline df0_2    = (DIC*1e-6*(K1 + 2.0*K1*K2)/(h_2**2.0 + K1*h_2 + K1*K2) - DIC*1e-6*(K1*h_2 + 2.0*K1*K2)/(h_2**2.0 + K1*h_2 + K1*K2)**2.0*(2.0*h_2 + K1) - TB*1.0/(1.0 + h_2/KB)**2.0/KB - KW/h_2**2.0 - 1.0/Free_2_T)*1e6*(-log(10.0)
```

```
    *10.0**(-pH_1))
inline pH_2     = pH_1 - f0_2/df0_2

// iteration 3

h_3             <- 10.0**(-pH_2)
h_free_3        <- h_3/Free_2_T
inline f0_3     = (DIC*1e-6*(K1*h_3 + 2.0*K1*K2)/(h_3*h_3 +
    K1*h_3 + K1*K2) - h_free_3 + KW/h_3 - alk*1e-6 + TB/(1.0 +
     h_3/KB))*1e6
inline df0_3    = (DIC*1e-6*(K1 + 2.0*K1*K2)/(h_3**2.0 + K1*
    h_3 + K1*K2) - DIC*1e-6*(K1*h_3 + 2.0*K1*K2)/(h_3**2.0 +
    K1*h_3 + K1*K2)**2.0*(2.0*h_3 + K1) - TB*1.0/(1.0 + h_3/KB
    )**2.0/KB - KW/h_3**2.0 - 1.0/Free_2_T)*1e6*(-log(10.0)
    *10.0**(-pH_2))
pH              <- pH_2 - f0_3/df0_3

// iteration 4

//       inline h_4      = 10.0**(-pH_3)
//       inline h_free_4 = h_4/Free_2_T
//       inline f0_4     = (DIC*1e-6*(K1*h_4 + 2.0*K1*K2)/(h_4
    *h_4 + K1*h_4 + K1*K2) - h_free_4 + KW/h_4 - alk*1e-6 + TB
    /(1.0 + h_4/KB))*1e6
//       inline df0_4    = (DIC*1e-6*(K1 + 2.0*K1*K2)/(h_4
    **2.0 + K1*h_4 + K1*K2) - DIC*1e-6*(K1*h_4 + 2.0*K1*K2)/(
    h_4**2.0 + K1*h_4 + K1*K2)**2.0*(2.0*h_4 + K1) - TB
    *1.0/(1.0 + h_4/KB)**2.0/KB - KW/h_4**2.0 - 1.0/Free_2_T)
    *1e6*(-log(10.0)*10.0**(-pH_3))
```

```
//       inline pH_4      = pH_3 - f0_4/df0_4

//       pH               <- pH_4

// calculate the final concentrations

inline H         = 10.0**(-pH)
inline H2        = H*H
inline denom     = (H2 + K1*H + K1*K2)
CO2              <- DIC*H2/denom
HCO3             <- DIC*H*K1/denom
CO3              <- DIC*K1*K2/denom

// end CO2SYS iterative solution


/* R and P as random walks */

r_R      ~ normal(0.0, sigma_r_R)
R        <- R + r_R
R1       <- exp(R)

r_P ~ normal(0.0, sigma_r_P)
P        <- P + r_P
P1       <- exp(P)

ode(h = 0.1, atoler = 1.0e-6, rtoler = 1.0e-6, alg = 'RK4(3)
   '){
```

```
dDIC/dt  =  -P1*24.0*I*mm + R1*24.0
           + gas*0.893*kLAO2*(CO2H - CO2)          + dil/V*(
   DIC_M - DIC)
dO_2/dt  =  (P1*24.0*I*mm - R1*24.0)/(RQ_d*I + RQ_n*(1.0-I))
      + gas*kLAO2*(O_2H - O_2)                + dil/V*(O_2_M
   - O_2)   + offset_O_2
dalk/dt  =   RR*P1*24.0*I*mm
                                                   + dil/V*(
   alk_M - alk)
dCp/dt   =   (P1*24.0*I*mm - R1*24.0)
                                                   + dil/V*(
   Cp)

}

mich_ment <- mm
O2H_pr    <- O_2H
CO2H_pr   <- CO2H

}


sub observation {

O2_obs   ~ log_normal(log(O_2), sigma_O_2)
pH_obs   ~ log_normal(log(pH),  sigma_pH)
DIC_obs ~ log_normal(log(DIC), sigma_DIC)
alk_obs ~ log_normal(log(alk), sigma_DIC)
}
```

```
}
```

LiBbi prior sampling file: prior.conf

```
--target prior
--model-file micro_iterative.bi
--nsamples 500
--start-time 0.61304
--end-time 4.7866
--noutputs 6049
--input-file data/input_all_2018_normalised.nc
--output-file results/prior_micro_iterative.nc
```

LiBbi posterior sampling file: posterior.conf

```
--target posterior
--model-file micro_iterative.bi
--input-file data/input_all_2018_normalised.nc
--obs-file data/obs_all_2018.nc
--nsamples 500
--nparticles 1024
--start-time 0.61304
--end-time 4.7866
--noutputs 6049
--output-file results/posterior_micro_iterative.nc
--with-transform-initial-to-param
```

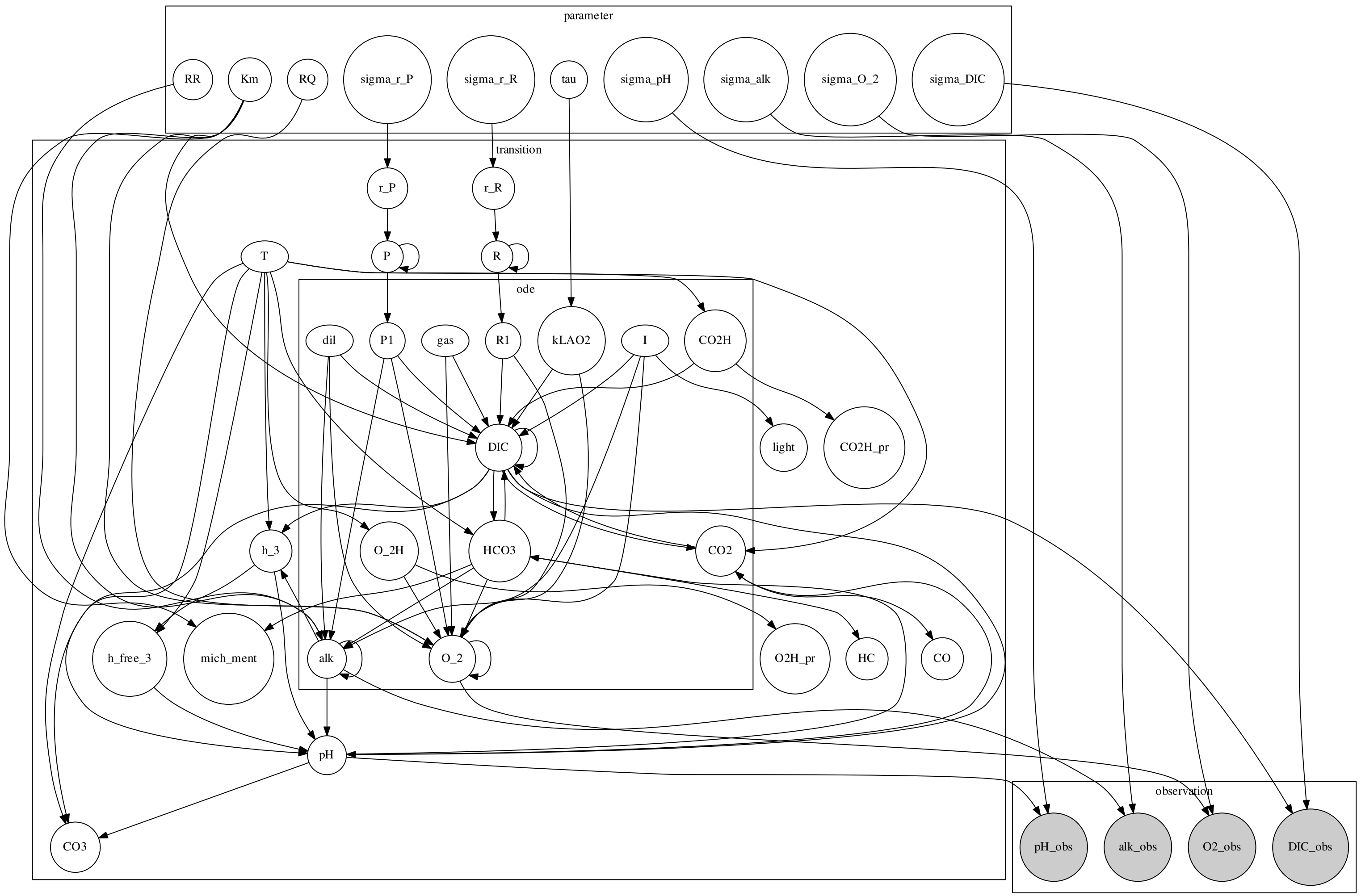

Figure A.1 : Directed Acyclic Graph of the LiBbi model file micro_iterative.bi

Bibliography

[1] Rubin Battino, Timothy R Rettich, and Toshihiro Tominaga. The solubility of oxygen and ozone in liquids. *Journal of physical and chemical reference data*, 12(2):163–178, 1983.

[2] Arnold E Greensberg, LS Clesceri, Andrew D Eaton, and MAH Franson. Standard methods for the examination of water and wastewater. *American Public Health Asociation, Whashington, DC*, 1992.

[3] E Molina Grima, JA Sánchez Pérez, F Ía Garc Camacho, and A Robles Medina. Gas-liquid transfer of atmospheric co2 in microalgal cultures. *Journal of Chemical Technology & Biotechnology*, 56(4):329–337, 1993.

[4] Robert RL Guillard and John H Ryther. Studies of marine planktonic diatoms: I. cyclotella nana hustedt, and detonula confervacea (cleve) gran. *Canadian journal of microbiology*, 8(2):229–239, 1962.

[5] Ernie Lewis, Doug Wallace, and Linda J Allison. Program developed for co {sub 2} system calculations. Technical report, Brookhaven National Lab., Dept. of Applied Science, Upton, NY (United States ..., 1998.

[6] Niels Ramsing and Jens Gundersen. Seawater and gases. *Limnol. Oceanogr*, 37:1307–1312, 2011.

[7] SMAC Van Heuven, D Pierrot, JWB Rae, E Lewis, and DWR Wallace. Matlab program developed for co2 system calculations. *ORNL/CDIAC-105b. Carbon Dioxide Information Analysis Center, Oak Ridge National Laboratory, US Department of Energy, Oak Ridge, Tennessee*, 530, 2011.

[8] R_F Weiss. Carbon dioxide in water and seawater: the solubility of a non-ideal gas. *Marine chemistry*, 2(3):203–215, 1974.