QUAL2K:

A Modeling Framework for Simulating River and Stream Water Quality

Documentation and Users Manual

The Mystic River at Medford, MA

Steve Chapra and Greg Pelletier

November 25, 2003

Chapra, S.C. and Pelletier, G.J. 2003. QUAL2K: A Modeling Framework for Simulating River and Stream Water Quality: Documentation and Users Manual. Civil and Environmental Engineering Dept., Tufts University, Medford, MA., Steven.Chapra@tufts.edu

1 INTRODUCTION

QUAL2K (or Q2K) is a river and stream water quality model that is intended to represent a modernized version of the QUAL2E (or Q2E) model (Brown and Barnwell 1987). Q2K is similar to Q2E in the following respects:

- One dimensional. The channel is well-mixed vertically and laterally.
- Steady state hydraulics. Non-uniform, steady flow is simulated.
- Diurnal heat budget. The heat budget and temperature are simulated as a function of meteorology on a diurnal time scale.
- Diurnal water-quality kinetics. All water quality variables are simulated on a diurnal time scale.
- Heat and mass inputs. Point and non-point loads and abstractions are simulated.

The QUAL2K framework includes the following new elements:

- Software Environment and Interface. Q2K is implemented within the Microsoft Windows environment. It is programmed in the Windows macro language: Visual Basic for Applications (VBA). Excel is used as the graphical user interface.
- Model segmentation. Q2E segments the system into river reaches comprised of equally spaced elements. In contrast, Q2K uses unequally-spaced reaches. In addition, multiple loadings and abstractions can be input to any reach.
- Carbonaceous BOD speciation. Q2K uses two forms of carbonaceous BOD to represent organic carbon. These forms are a slowly oxidizing form (slow CBOD) and a rapidly oxidizing form (fast CBOD). In addition, non-living particulate organic matter (detritus) is simulated. This detrital material is composed of particulate carbon, nitrogen and phosphorus in a fixed stoichiometry.
- Anoxia. Q2K accommodates anoxia by reducing oxidation reactions to zero at low oxygen levels. In addition, denitrification is modeled as a first-order reaction that becomes pronounced at low oxygen concentrations.
- Sediment-water interactions. Sediment-water fluxes of dissolved oxygen and nutrients are simulated internally rather than being prescribed. That is, oxygen (SOD) and nutrient fluxes are simulated as a function of settling particulate organic matter, reactions within the sediments, and the concentrations of soluble forms in the overlying waters.
- Bottom algae. The model explicitly simulates attached bottom algae.
- Light extinction. Light extinction is calculated as a function of algae, detritus and inorganic solids.
- pH. Both alkalinity and total inorganic carbon are simulated. The river's pH is then simulated based on these two quantities.
- Pathogens. A generic pathogen is simulated. Pathogen removal is determined as a function of temperature, light, and settling.

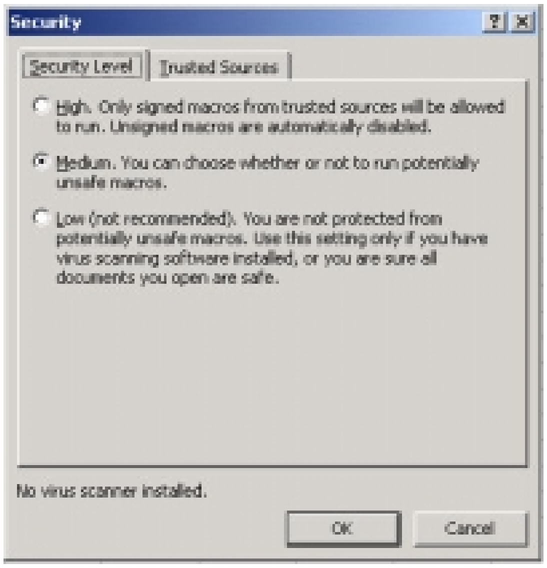
2 GETTING STARTED

Installation is required for many water-quality models. This is not the case for QUAL2K because the model is packaged as an Excel Workbook. The program is written in Excel's macro language: Visual Basic for Applications or VBA. The Excel Workbook's worksheets and charts are used to enter data and display results. Consequently, you merely have to open the Workbook to begin modeling. The following are some recommended step-by-step instructions on how to obtain your first model run.

Step 1: Create a folder named QUAL2K to hold the workbook and its data files. For example, in the following example, a folder named QUAL2K is created on the C:\ drive.

Step 2: Copy the Q2KMaster file (Q2KMaster.xls) from your CD to C:\QUAL2K.

Step 3: Open Excel and make sure that your macro security level is set to medium (Figure 1). This can be done using the menu commands: Tools → Macro → Security. Make certain that the Medium radio button is selected.



○ High. Only signed macros from trusted sources will be allowed to run. Unsigned macros are automatically disabled.

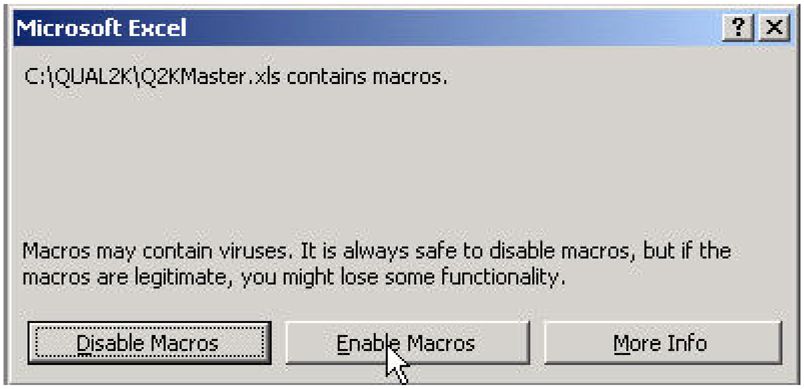
● Medium. You can choose whether or not to run potentially unsafe macros.

○ Low (not recommended). You are not protected from potentially unsafe macros. Use this setting only if you have virus scanning software installed, or you are sure all documents you open are safe.

No virus scanner installed.

Figure 1 The Excel Macro Security Level dialogue box. In order to run Q2K, the Medium level of security should be selected.

Step 4: Open Q2KMaster.xls. When you do this, the Macro Security Dialogue Box will be displayed (Figure 2).



Microsoft Excel

C:\QUAL2K\Q2KMaster.xls contains macros.

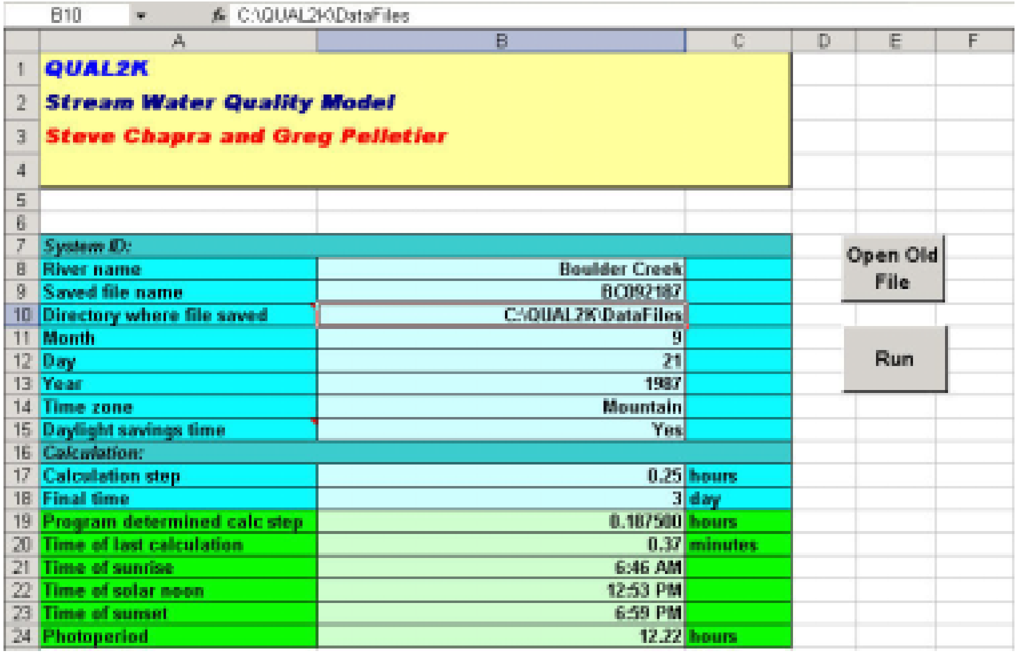
Macros may contain viruses. It is always safe to disable macros, but if the macros are legitimate, you might lose some functionality.

Figure 2 The Excel Macro security dialogue box. In order to run Q2K, the Enable Macros button must be selected.

Click on the Enable Macros button.

Step 5: Immediately save the file as Q2K.xls. This will be the Excel Workbook that you will use on a routine basis. If for some reason, you modify Q2K.xls in a way that makes it unusable, you can always go back to Q2KMaster.xls as a backup.

Step 6: On the QUAL2K worksheet, go to cell B10 and enter the path to the DataFiles directory: C:\QUAL2K\DataFiles as shown in Figure 3.



B10 C:\QUAL2K\DataFiles

	A	B	C
1	QUAL2K		
2	Stream Water Quality Model		
3	Steve Chapra and Greg Pelletier		
4			
5			
6			
7	System ID:		
8	River name	Boulder Creek	
9	Saved file name	BC092187	
10	Directory where file saved	C:\QUAL2K\DataFiles	
11	Month	9	
12	Day	21	
13	Year	1987	
14	Time zone	Mountain	
15	Daylight savings time	Yes	
16	Calculation:		
17	Calculation step	0.25	hours
18	Final time	3	day
19	Program determined calc step	0.187500	hours
20	Time of last calculation	0.37	minutes
21	Time of sunrise	6:46 AM	
22	Time of solar noon	12:53 PM	
23	Time of sunset	6:59 PM	
24	Photoperiod	12.22	hours

Figure 3 The QUAL2K worksheet showing the entry of the file path into cell B10.

Step 7: Click on the Run button. QUAL2K will begin to execute. You can follow the progress of the execution on the status bar that is shown in the bottom left corner of the worksheet (Figure 4).

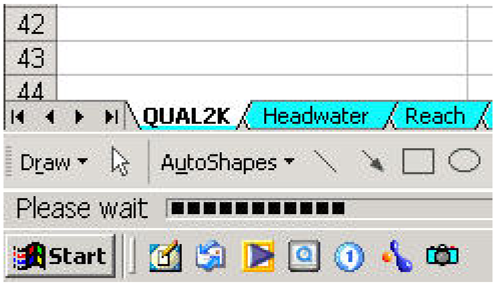
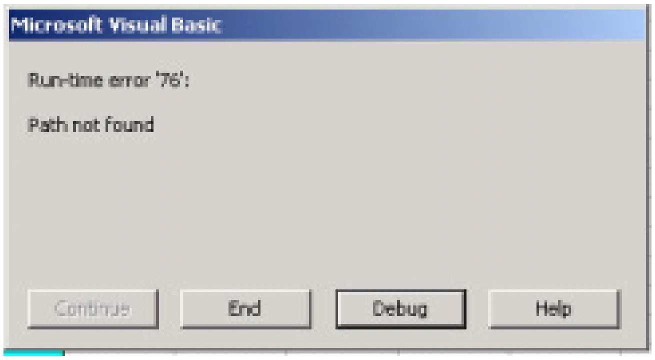


Figure 4 The QUAL2K Status Bar is positioned at the lower left corner of the worksheet. It allows you to follow the progress of a model run.

If the program runs properly, the temperature plot will be displayed. If it does not work properly, two possibilities exist:

First, you may be using an old version of Microsoft Office. Although Excel is downwardly compatible for some earlier versions, Q2K will not work with very old versions.

Second, you may have made a mistake in implementing the preceding steps. A common mistake is to have mistyped the file path that you entered in cell B10. If this is the case, you will receive an error message (Figure 5).



Run-time error '76':

Path not found

Figure 5 An error message that will occur if you type the incorrect file path into cell B10 on the QUAL2K worksheet.

If this occurs, click on the end button. This will terminate the run and bring you back to the Excel Workbook. You should then move back to the QUAL2K worksheet and correct the file path entry. Note that the same error will occur if you did not set up your directories with correct names as specified above.

Step 8: On the QUAL2K worksheet click on the Open Old File button. Browse to get to the directory: C:\QUAL2K\DataFiles. You should see that a new file: BC092187.q2k has been created. Click on the Cancel button to return to Q2K.

Note that every time that Q2K is run, a data file will be created with the file name specified in cell B9 on the QUAL2K worksheet (Figure 3). The program automatically affixes the extension .q2k to the file name. Since this will overwrite the file, make certain to change the file name when you perform a new application.

3 SEGMENTATION AND HYDRAULICS

The model presently simulates the main stem of a river as depicted in Figure 6. Tributaries are not modeled explicitly, but can be represented as point sources.

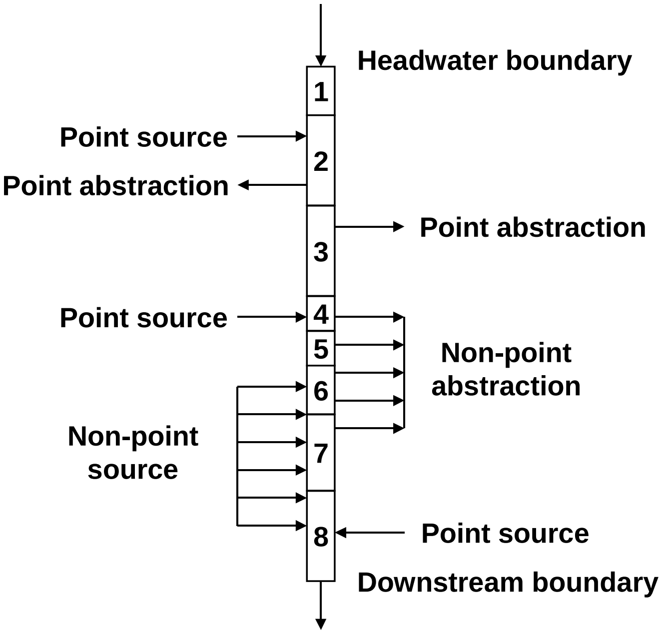


Figure 6 QUAL2K segmentation scheme.

3.1 Flow Balance

A steady-state flow balance is implemented for each model reach (Figure 7)

$$Q_i = Q_{i-1} + Q_{in,i} - Q_{ab,i} \tag{1}$$

where Q_i = outflow from reach i into reach i + 1 [m^3/d], Q_{i-1} = inflow from the upstream reach i – 1 [m^3/d], $Q_{in,i}$ is the total inflow into the reach from point and nonpoint sources [m^3/d], and $Q_{ab,i}$ is the total outflow from the reach due to point and nonpoint abstractions [m^3/d].

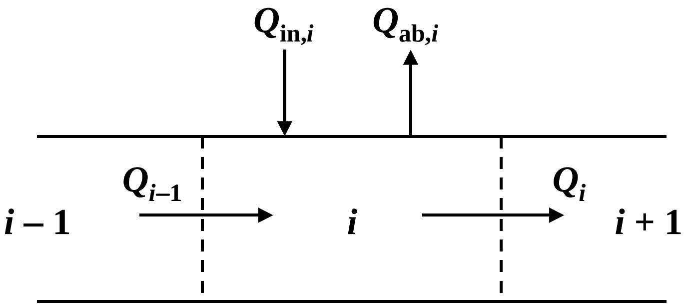


Figure 7 Reach flow balance.

The total inflow from sources is computed as

$$Q_{in,i} = \sum_{j=1}^{psi} Q_{ps,i,j} + \sum_{j=1}^{npsi} Q_{nps,i,j} \tag{2}$$

where $Q_{ps,i,j}$ is the jth point source inflow to reach i [m^3/d], psi = the total number of point sources to reach i, $Q_{nps,i,j}$ is the jth non-point source inflow to reach i [m^3/d], and $npsi$ = the total number of non-point source inflows to reach i.

The total outflow from abstractions is computed as

$$Q_{ab,i} = \sum_{j=1}^{pai} Q_{pa,i,j} + \sum_{j=1}^{npai} Q_{npa,i,j} \tag{3}$$

where $Q_{pa,i,j}$ is the jth point abstraction outflow from reach i [m^3/d], pai = the total number of point abstractions from reach i, $Q_{npa,i,j}$ is the jth non-point abstraction outflow from reach i [m^3/d], and $npai$ = the total number of non-point abstraction flows from reach i.

The non-point sources and abstractions are modeled as line sources. As in Figure 8, the non-point source or abstraction is demarcated by its starting and ending kilometer points. Its flow is distributed to or from each reach in a length-weighted fashion.

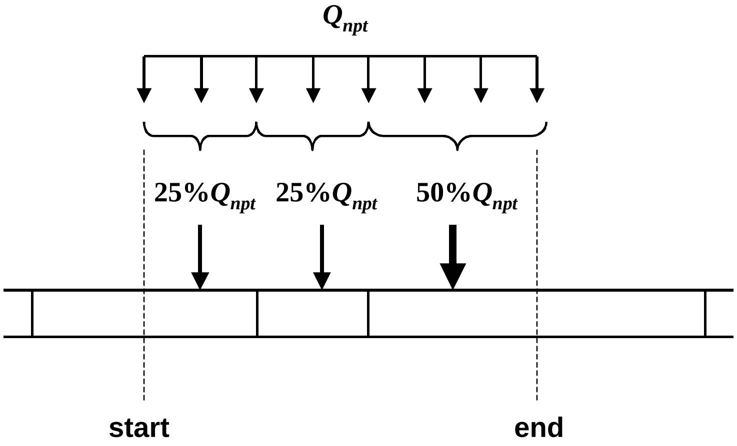


Figure 8 The manner in which non-point source flow is distributed to a reach.

3.2 Hydraulic Characteristics

Once the outflow for each reach is computed, the depth and velocity are calculated in one of three ways: weirs, rating curves, and Manning equations. The program decides among these options in the following manner:

- If a weir height is entered, the weir option is implemented.

- If the weir height is zero and a roughness coefficient is entered (*n*), the Manning equation option is implemented.
- If neither of the previous conditions are met, Q2K uses rating curves.

3.2.1 Weirs

Figure 9 shows how weirs are represented in Q2K. The symbols are defined as: H_i = the depth of the reach upstream of the weir [m], H_{i+1} = the depth of the reach downstream of the weir [m], $elev2_i$ = the elevation above sea level of the upstream reach [m], $elev1_{i+1}$ = the elevation above sea level of the downstream reach [m], H_w = the height of the weir above elev2$_i$ [m], H_d = the drop between the elevation above sea level of the surface of reach *i* and reach *i*+1 [m], H_h = the head above the weir [m], B_i = the width of reach *i* [m].

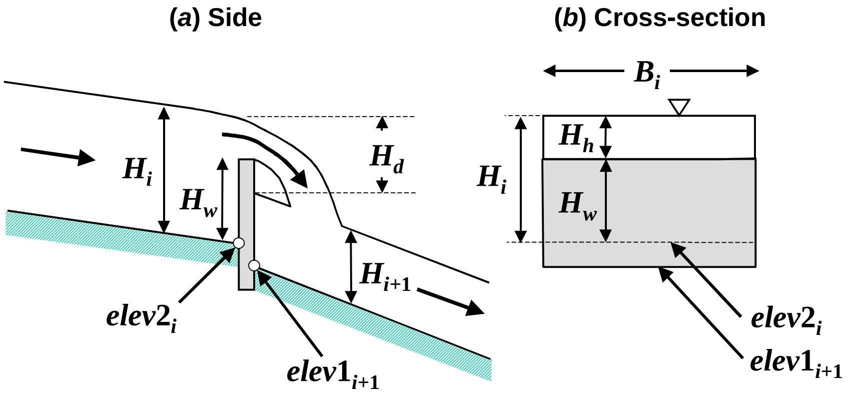


Figure 9 A sharp-crested weir.

For a sharp-crested weir where $H_h/H_w < 0.4$, flow is related to head by (Finnemore and Franzini 2002)

$$Q_i = 1.83 B_i H_h^{3/2} \tag{4}$$

where Q_i is the outflow from the segment upstream of the weir in m^3/s, and B_i and H_h are in m. Equation (4) can be solved for

$$H_h = \left(\frac{Q_i}{1.83 B_i} \right)^{2/3} \tag{5}$$

This result can then be used to compute the depth of reach *i,* and the drop over the weir

$$H_i = H_w + H_h \tag{6}$$

and

$$H_d = elev2_i + H_i - elev1_{i+1} - H_{i+1} \tag{7}$$

The velocity and cross-sectional area of reach *i* can then be computed as

$$A_{c,i} = B_i H_i \tag{8}$$

$$U_i = \frac{Q_i}{A_{c,i}} \tag{9}$$

3.2.2 Rating Curves

Power equations can be used to relate mean velocity and depth to flow,

$$U = aQ^b \tag{10}$$

$$H = \alpha Q^\beta \tag{11}$$

where a, b, α and β are empirical coefficients that are determined from velocity-discharge and stage-discharge rating curves, respectively. The values of velocity and depth can then be employed to determine the cross-sectional area and width by

$$A_c = \frac{Q}{U} \tag{12}$$

$$B = \frac{A_c}{H} \tag{13}$$

The exponents b and β typically take on values listed in Table 1. Note that the sum of b and β must be less than or equal to 1. If their sum equals 1, the channel is rectangular.

Table 1 Typical values for the exponents of rating curves used to determine velocity and depth from flow (Barnwell et al. 1989).

Equation	Exponent	Typical value	Range
$U = aQ^b$	b	0.43	0.4–0.6
$H = \alpha Q^\beta$	β	0.45	0.3–0.5

3.2.3 Manning Equation

Each reach is idealized as a trapezoidal channel (Figure 10). Under conditions of steady flow, the Manning equation can be used to express the relationship between flow and depth as

$$Q = \frac{S_0^{1/2}}{n} \frac{A_c^{5/3}}{P^{2/3}} \tag{14}$$

where Q = flow [m^3/s][1], S_0 = bottom slope [m/m], n = the Manning roughness coefficient, A_c = the cross-sectional area [m^2], and P = the wetted perimeter [m].

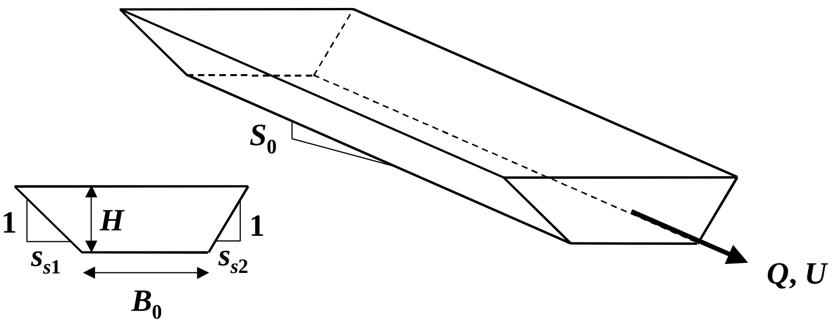


Figure 10 Trapezoidal channel.

The cross-sectional area of a trapezoidal channel is computed as

$$A_c = [B_0 + 0.5(s_{s1} + s_{s2})H]H \tag{15}$$

where B_0 = bottom width [m], s_{s1} and s_{s2} = the two side slopes as shown in Figure 10 [m/m], and H = reach depth [m].

The wetted perimeter is computed as

$$P = B_0 + H\sqrt{s_{s1}^2 + 1} + H\sqrt{s_{s2}^2 + 1} \tag{16}$$

After substituting Equations (15) and (16), Equation (14) can be solved iteratively for depth (Chapra and Canale 2002),

$$H_{k+1} = \frac{(Qn)^{3/5}\left(B_0 + H\sqrt{s_{s1}^2 + 1} + H\sqrt{s_{s2}^2 + 1}\right)^{2/5}}{S^{3/10}[B_0 + 0.5(s_{s1} + s_{s2})H]} \tag{17}$$

where k = 0, 1, 2, …, n, where n = the number of iterations. The method is terminated when the estimated error falls below a specified value of 0.001%. The estimated error is calculated as

$$\varepsilon_a = \left|\frac{H_{k+1} - H_k}{H_{k+1}}\right| \times 100\% \tag{18}$$

The cross-sectional area can be determined with Equation (15) and the velocity can then be determined from the continuity equation,

---

[1] Notice that time is measured in seconds in this and other formulas used to characterize hydraulics. This is how the computations are implemented within Q2K. However, once the hydraulic characteristics are determined they are converted to day units to be compatible with other computations.

$$U = \frac{Q}{A_c} \tag{19}$$

The average reach width, B [m], can be computed as

$$B = \frac{A_c}{H} \tag{20}$$

Suggested values for the Manning coefficient are listed in Table 2.

Table 2 The Manning roughness coefficient for various open channel surfaces (from Chow et al. 1988).

MATERIAL	*n*
Man-made channels	
Concrete	0.012
Gravel bottom with sides:	
concrete	0.020
mortared stone	0.023
riprap	0.033
Natural stream channels	
Clean, straight	0.025-0.04
Clean, winding and some weeds	0.03-0.05
Weeds and pools, winding	0.05
Mountain streams with boulders	0.04-0.10
Heavy brush, timber	0.05-0.20

Manning's n typically varies with flow and depth (Gordon et al, 1992). As the depth decreases at low flow, the relative roughness increases. Typical published values of Manning's n, which range from about 0.015 for smooth channels to about 0.15 for rough natural channels, are representative of conditions when the flow is at the bankfull capacity (Rosgen, 1996). Critical conditions of depth for evaluating water quality are generally much less than bankfull depth, and the relative roughness may be much higher.

3.2.4 Waterfalls

In Section 3.2.1, the drop of water over a weir was computed. This value is needed in order to compute the enhanced reaeration that occurs in such cases. In addition to weirs, such drops can also occur at waterfalls (Figure 11).

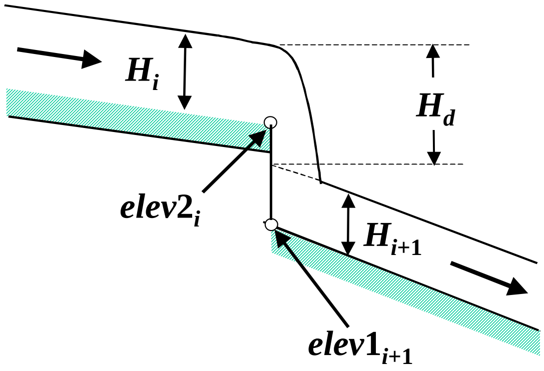


Figure 11 Waterfall.

QUAL2K computes such drops for cases where the elevation above sea level drops abruptly at the boundary between two reaches. For both the rating curve and the Manning equation options, the model uses Equation (7) for this purpose. It should be noted that the drop is only calculated when the elevation above sea level at the downstream end of a reach is greater than at the beginning of the next downstream reach; that is, $elev2_i > elev1_{i+1}$.

3.3 Travel Time

The residence time of each reach is computed as

$$\tau_k = \frac{V_k}{Q_k} \tag{21}$$

where τ_k = the residence time of the k^{th} reach [d], V_k = the volume of the k^{th} reach [m^3] = $A_{c,k}\Delta x_k$, and Δx_k = the length of the k^{th} reach [m]. These times are then accumulated to determine the travel time from the headwater to the downstream end of reach i,

$$t_{t,i} = \sum_{k=1}^{i} \tau_k \tag{22}$$

where $t_{t,i}$ = the travel time [d].

$$\tau_k = \frac{V_k}{Q_k}$$

$$t_{t,i} = \sum_{k=1}^{i} \tau_k$$

3.4 Longitudinal Dispersion

Two options are used to determine the longitudinal dispersion for a boundary between two reaches. First, the user can simply enter estimated values. If the user does not enter values, a hydraulics based formula is employed to internally compute dispersion based on the channel's hydraulics (Fischer et al. 1979),

$$E_{p,i}\,0.011\frac{U_i^2 B_i^2}{H_i U_i^*} \tag{23}$$

where $E_{p,i}$ = the longitudinal dispersion between reaches i and $i + 1$ [m^2/s], U_i = velocity [m/s], B_i = width [m], H_i = mean depth [m], and U_i^* = shear velocity [m/s], which is related to more fundamental characteristics by

$$U_i^* = \sqrt{gH_i S_i} \tag{24}$$

where g = acceleration due to gravity [= 9.81 m/s^2] and S = channel slope [dimensionless].

After computing or prescribing $E_{p,i}$, the numerical dispersion is computed as

$$E_{n,i} = \frac{U_i \Delta x_i}{2} \tag{25}$$

The model dispersion E_i (i.e., the value used in the model calculations) is then computed as follows:

- If $E_{n,i} \le E_{p,i}$, the model dispersion, E_i is set to $E_{p,i} - E_{n,i}$.
- If $E_{n,i} > E_{p,i}$, the model dispersion is set to zero.

For the latter case, the resulting dispersion will be greater than the physical dispersion. Thus, dispersive mixing will be higher than reality. It should be noted that for most steady state rivers, the impact of this overestimation on concentration gradients will be negligible. If the discrepancy is significant, the only alternative is to make reach lengths smaller so that the numerical dispersion becomes smaller than the physical dispersion.

4 TEMPERATURE MODEL

As in Figure 12, the heat balance takes into account heat transfers from adjacent reaches, loads, abstractions, the atmosphere, and the sediments. A heat balance can be written for reach *i* as

$$\begin{aligned}\frac{dT_i}{dt} = \frac{Q_{i-1}}{V_i}T_{i-1} - \frac{Q_i}{V_i}T_i - \frac{Q_{ab,i}}{V_i}T_i + \frac{E'_{i-1}}{V_i}\left(T_{i-1} - T_i\right) + \frac{E'_i}{V_i}\left(T_{i+1} - T_i\right) \\ + \frac{W_{h,i}}{\rho_w C_{pw} V_i}\left(\frac{\text{m}^3}{10^6\ \text{cm}^3}\right) + \frac{J_{h,i}}{\rho_w C_{pw} H_i}\left(\frac{\text{m}}{100\,\text{cm}}\right) + \frac{J_{s,i}}{\rho_w C_{pw} H_i}\left(\frac{\text{m}}{100\,\text{cm}}\right)\end{aligned} \tag{26}$$

where T_i = temperature in reach *i* [$^{\text{o}}$C], *t* = time [d], E'_i = the bulk dispersion coefficient between reaches *i* and *i* + 1 [m^3/d], $W_{h,i}$ = the net heat load from point and non-point sources into reach *i* [cal/d], ρ_w = the density of water [g/cm^3], C_{pw} = the specific heat of water [cal/(g $^{\text{o}}$C)], $J_{h,i}$ = the air-water heat flux [cal/(cm^2 d)], and $J_{s,i}$ = the sediment-water heat flux [cal/(cm^2 d)].

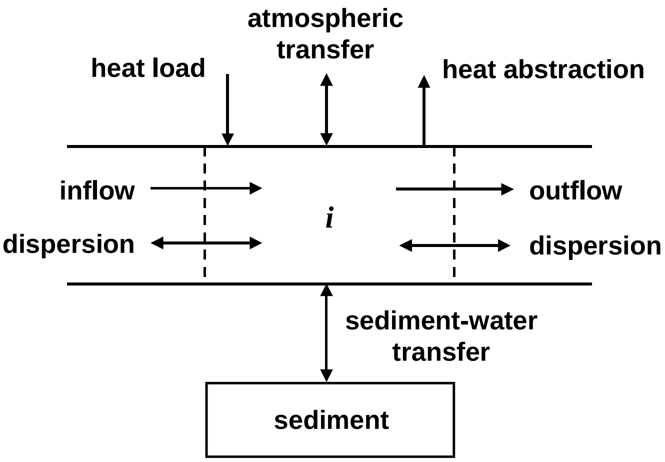


Figure 12 Heat balance.

The bulk dispersion coefficient is computed as

$$E'_i = \frac{E_i A_{c,i}}{\left(\Delta x_i + \Delta x_{i+1}\right)/2} \tag{27}$$

Note that a zero dispersion condition is applied at the downstream boundary.

The net heat load from sources is computed as

$$W_{h,i} = \rho C_p \left[\sum_{j=1}^{psi} Q_{ps,i,j} T_{psi,j} + \sum_{j=1}^{npsi} Q_{nps,i,j} T_{npsi,j}\right] \tag{28}$$

where $T_{ps,i,j}$ is the *j*th point source temperature for reach *i* [^{o}C], and $T_{nps,i,j}$ is the *j*th non-point source temperature for reach *i* [^{o}C].

4.1 Surface Heat Flux

As depicted in Figure 13, surface heat exchange is modeled as a combination of five processes:

$$J_h = I(0) + J_{an} - J_{br} - J_c - J_e \tag{29}$$

where $I(0)$ = net solar shortwave radiation at the water surface, J_{an} = net atmospheric longwave radiation, J_{br} = longwave back radiation from the water, J_c = conduction, and J_e = evaporation. All fluxes are expressed as cal/cm^2/d.

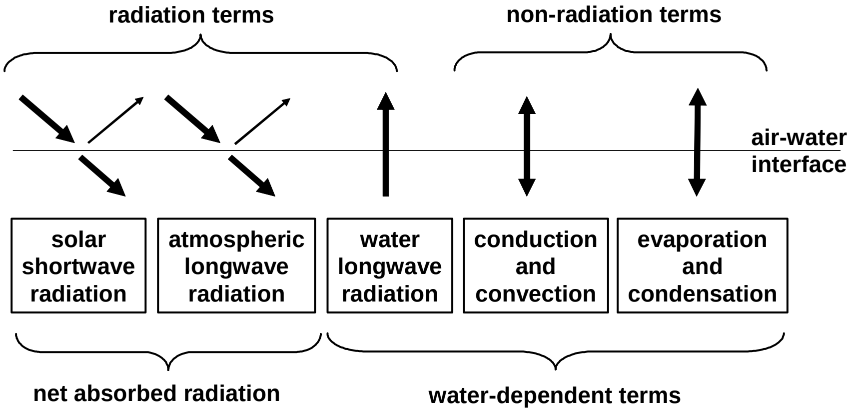


Figure 13 The components of surface heat exchange.

4.1.1 Solar Radiation

The model computes the amount of solar radiation entering the water at a particular latitude (L_{at}) and longitude (L_{lm}) on the earth's surface. This quantity is a function of the radiation at the top of the earth's atmosphere which is attenuated by atmospheric transmission, cloud cover, shade, and reflection,

$$I(0) = \underbrace{I_0}_{\substack{\text{extraterrestrial}\\ \text{radiation}}} \quad \underbrace{a_t}_{\substack{\text{atmospheric}\\ \text{attenuation}}} \quad \underbrace{a_c}_{\substack{\text{cloud}\\ \text{attenuation}}} \quad \underbrace{(1-R_s)}_{\text{reflection}} \quad \underbrace{(1-S_f)}_{\text{shading}} \tag{30}$$

where $I(0)$ = solar radiation at the water surface [cal/cm^2/d], I_0 = extraterrestrial radiation (i.e., at the top of the earth's atmosphere) [cal/cm^2/d], a_t = atmospheric attenuation, C_L = fraction of sky covered with clouds, R_s = albedo (fraction reflected), and S_f = effective shade (fraction blocked by vegetation and topography).

Extraterrestrial radiation. The extraterrestrial radiation is computed as (TVA 1972)

$$I_0 = \frac{W_0}{r^2} \sin \alpha \tag{31}$$

where W_0 = the solar constant [1367 W/m^2 or 2823 cal/cm^2/d], r = normalized radius of the earth's orbit (i.e., the ratio of actual earth-sun distance to mean earth-sun distance), and α = the sun's altitude [radians], which can be computed as

$$\sin \alpha = \sin \delta \sin L_{at} + \cos \delta \cos L_{at} \cos(\tau) \tag{32}$$

where δ = solar declination [radians], L_{at} = local latitude [radians], and τ = the local hour angle of the sun [radians].

The normalized radius can be estimated as

$$r = 1 + 0.017 \cos\left(\frac{2\pi}{365}(186 - Dy)\right) \tag{33}$$

where Dy = Julian day (Jan. 1 = 1, Jan. 2 = 2, etc.).

The solar declination can be estimated as

$$\delta = 23.45 \frac{\pi}{180} \cos\left(\frac{2\pi}{365}(172 - Dy)\right) \tag{34}$$

The local hour angle in radians is given by

$$\tau = \left(\frac{trueSolarTime}{4} - 180\right)\frac{\pi}{180} \tag{35}$$

where:

$$trueSolarTime = localTime + eqtime - 4 \times L_{lm} - 60 \times timezone \tag{36}$$

where *trueSolarTime* is the solar time determined from the actual position of the sun in the sky [minutes], *localTime* is the local time in minutes (local standard time), L_{lm} is the local longitude (positive decimal degrees for the western hemisphere), and *timezone* is the local time zone in hours relative to Greenwich Mean Time (e.g. -8 hours for Pacific Standard Time; the local time zone is selected on the **Qual2K** worksheet). The value of *eqtime* represents the difference between true solar time and mean solar time in minutes of time.

QUAL2K calculates the sun's altitude and *eqtime*, as well as the times of sunrise and sunset using the Meeus (1999) algorithms as implemented by NOAA's Surface Radiation Research Branch (www.srrb.noaa.gov/highlights/sunrise/azel.html). The NOAA method for solar position that is used in QUAL2K also includes a correction for the effect of atmospheric refraction. The complete calculation method that is used to determine the solar position, *eqtime*, sunrise and sunset is presented in Appendix 2.

The photoperiod f [hours] is computed as

$$f = t_{ss} - t_{sr} \tag{37}$$

where t_{ss} = time of sunset [hours] and t_{sr} = time of sunrise [hours].

Atmospheric attenuation. Various methods have been published to estimate the fraction of the atmospheric attenuation from a clear sky (a_t). Two alternative methods are available in QUAL2K to estimate a_t:

- Bras (1990)
- Ryan and Stolzenbach (1972)

Note that the solar radiation model is selected on the **Light and Heat** worksheet of QUAL2K.

The Bras (1990) method computes a_t as:

$$a_t = e^{-n_{fac} a_1 m} \tag{38}$$

where n_{fac} is an atmospheric turbidity factor that varies from approximately 2 for clear skies to 4 or 5 for smoggy urban areas. The molecular scattering coefficient (a_1) is calculated as

$$a_1 = 0.128 - 0.054 \log_{10} m \tag{39}$$

where m is the optical air mass, calculated as

$$m = \frac{1}{\sin\alpha + 0.15(\alpha_d + 3.885)^{-1.253}} \tag{40}$$

where α_d is the sun's altitude in degrees from the horizon = $\alpha \times (180^\circ/\pi)$.

The Ryan and Stolzenbach (1972) model computes a_t from ground surface elevation and solar altitude as:

$$a_t = a_{tc}^{\,m\left(\frac{288 - 0.0065 elev}{288}\right)^{5.256}} \tag{41}$$

where a_{tc} is the atmospheric transmission coefficient (0.70-0.91, typically approximately 0.8), and $elev$ is the ground surface elevation in meters.

Direct measurements of solar radiation are available at some locations. For example, NOAA's Integrated Surface Irradiance Study (ISIS) has data from various stations across the United States (http://www.atdd.noaa.gov/isis.htm). The selection of either the Bras or Ryan-Stolzenbach solar radiation model and the appropriate atmospheric turbidity factor or atmospheric transmission coefficient for a particular application should ideally be guided by a comparison of predicted solar radiation with measured values at a reference location.

Cloud Attenuation. Attenuation of solar radiation due to cloud cover is computed with

$$a_c = 1 - 0.65C_L^2 \tag{42}$$

where C_L = fraction of the sky covered with clouds.

Reflectivity. Reflectivity is calculated as

$$R_s = A\alpha_d{}^B \tag{43}$$

where A and B are coefficients related to cloud cover (Table 3).

Table 3 Coefficients used to calculate reflectivity based on cloud cover.

Cloudiness	Clear		Scattered		Broken		Overcast	
C_L	0		0.1-0.5		0.5-0.9		1.0	
Coefficients	*A*	*B*	*A*	*B*	*A*	*B*	*A*	*B*
	1.18	–0.77	2.20	–0.97	0.95	–0.75	0.35	–0.45

Shade. Shade is an input variable for the QUAL2K model. Shade is defined as the fraction of potential solar radiation that is blocked by topography and vegetation. An Excel/VBA program named ‘Shade.xls’ is available from the Washington Department of Ecology to estimate shade from topography and riparian vegetation (Ecology 2003). Input values of integrated hourly estimates of shade are put into the **Shade** worksheet of QUAL2K.

4.1.2 Atmospheric Long-wave Radiation

The downward flux of longwave radiation from the atmosphere is one of the largest terms in the surface heat balance. This flux can be calculated using the Stefan-Boltzmann law

$$J_{an} = \sigma(T_{air} + 273)^4 \quad \varepsilon_{sky} \quad (1 - R_L) \tag{44}$$

where σ = the Stefan-Boltzmann constant = 11.7x10^{-8} cal/(cm^2 d K^4), T_{air} = air temperature [^{o}C], ε_{sky} = effective emissivity of the atmosphere [dimensionless], and R_L = longwave reflection coefficient [dimensionless]. Emissivity is the ratio of the longwave radiation from an object compared with the radiation from a perfect emitter at the same temperature. The reflection coefficient is generally small and in typically assumed to equal 0.03.

The atmospheric longwave radiation model is selected on the **Light and Heat** worksheet of QUAL2K. Three alternative methods are available for use in QUAL2K to represent the effective emissivity (ε_{sky}):

Brutsaert (1982). The Brutsaert equation is physically-based instead of empirically derived and has been shown to yield satisfactory results over a wide range of atmospheric conditions of air temperature and humidity at intermediate latitudes for conditions above freezing (Brutsaert, 1982).

$$\varepsilon_{clear} = 1.24\left(\frac{1.333224e_{air}}{T_a}\right)^{1/7}$$

where e_{air} is the air vapor pressure [mm Hg], and T_a is the air temperature in °K. The factor of 1.333224 converts the vapor pressure from mm Hg to millibars. The air vapor pressure [in mm Hg] is computed as (Raudkivi 1979):

$$e_{air} = 4.596 e^{\frac{17.27 T_d}{237.3 + T_d}} \tag{45}$$

where T_d = the dew-point temperature [^{o}C].

Brunt (1932). Brunt's equation is an empirical model that has been commonly used in water-quality models (Thomann and Mueller 1987),

$$\varepsilon_{clear} = A_a + A_b \sqrt{e_{air}}$$

where A_a and A_b are empirical coefficients. Values of A_a have been reported to range from about 0.5 to 0.7 and values of A_b have been reported to range from about 0.031 to 0.076 mmHg$^{-0.5}$ for a wide range of atmospheric conditions. QUAL2K uses a default mid-range value of A_a = 0.6 with A_b = 0.031 mmHg$^{-0.5}$ if the Brunt method is selected on the **Light and Heat** worksheet.

Koberg (1964). Koberg (1964) reported that the A_a in Brunt's formula depends on both air temperature and the ratio of the incident solar radiation to the clear-sky radiation (R_{sc}). As in Figure 14, he presented a series of curves indicating that A_a increases with T_{air} and decreases with R_{sc} with A_b held constant at 0.0263 millibars$^{-0.5}$ (about 0.031 mmHg$^{-0.5}$).

The following polynomial is used in Q2K to provide a continuous approximation of Koberg's curves.

$$A_a = a_k T_{air}^2 + b_k T_{air} + c_k$$

where

$$a_k = -0.00076437 R_{sc}^3 + 0.00121134 R_{sc}^2 - 0.00073087 R_{sc} + 0.0001106$$
$$b_k = 0.12796842\, R_{sc}^3 - 0.2204455 R_{sc}^2 + 0.13397992\, R_{sc} - 0.02586655$$
$$c_k = -3.25272249 R_{sc}^3 + 5.65909609 R_{sc}^2 - 3.43402413 R_{sc} + 1.43052757$$

The fit of this polynomial to points sampled from Koberg's curves are depicted in Figure 14. Note that an upper limit of 0.735 is prescribed for A_a.

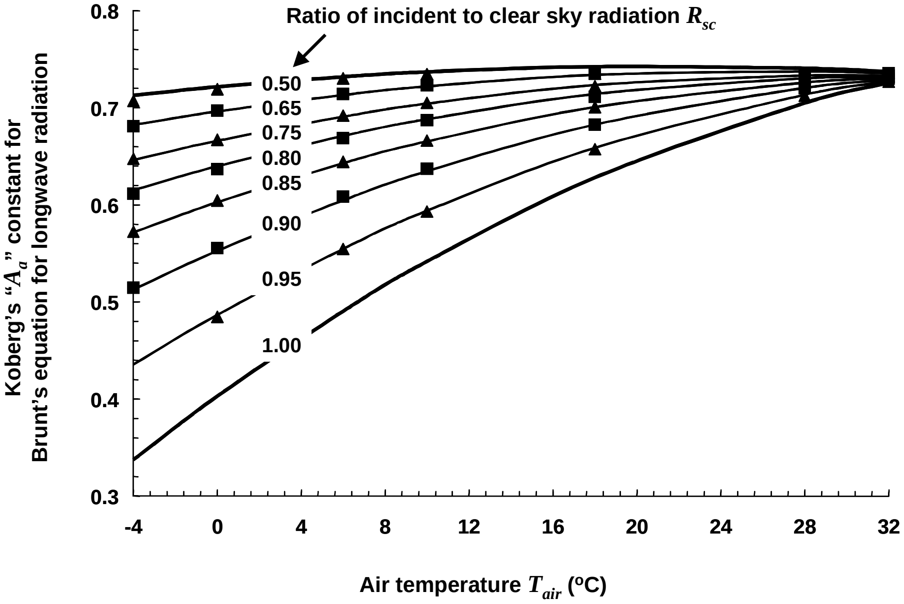


Figure 14 The points are sampled from Koberg's family of curves for determining the value of the A_a constant in Brunt's equation for atmospheric longwave radiation (Koberg, 1964). The lines are the functional representation used in Q2K.

For cloudy conditions the atmospheric emissivity may increase as a result of increased water vapor content. High cirrus clouds may have a negligible effect on atmospheric emissivity, but lower stratus and cumulus clouds may have a significant effect. The Koberg method accounts for the effect of clouds on the emissivity of longwave radiation in the determination of the A_a coefficient. The Brunt and Brutsaert methods determine emissivity of a clear sky and do not account for the effect of clouds. Therefore, if the Brunt or Brutsaert methods are selected, then the effective atmospheric emissivity for cloudy skies (ε_{sky}) is estimated from the clear sky emissivity by using a nonlinear function of the fractional cloud cover (C_L) of the form (TVA, 1972):

$$\varepsilon_{sky} = \varepsilon_{clear}(1 + 0.17C_L^2) \qquad \textbf{(46)}$$

The selection of the longwave model for a particular application should ideally be guided by a comparison of predicted results with measured values at a reference location. However, direct measurements are rarely collected. The Brutsaert method is recommended to represent a wide range of atmospheric conditions.

4.1.3 Water Long-wave Radiation

The back radiation from the water surface is represented by the Stefan-Boltzmann law,

$$J_{br} = \varepsilon\sigma(T + 273)^4 \qquad \textbf{(47)}$$

where ε = emissivity of water (= 0.97) and T = the water temperature [°C].

4.1.4 Conduction and Convection

Conduction is the transfer of heat from molecule to molecule when matter of different temperatures are brought into contact. Convection is heat transfer that occurs due to mass movement of fluids. Both can occur at the air-water interface and can be described by,

$$J_c = c_1 f(U_w)(T_s - T_{air}) \tag{48}$$

where c_1 = Bowen's coefficient (= 0.47 mmHg/^{o}C). The term, $f(U_w)$, defines the dependence of the transfer on wind velocity over the water surface where U_w is the wind speed measured a fixed distance above the water surface.

Many relationships exist to define the wind dependence. Bras (1990), Edinger et al (1974), and Shanahan (1984) provide reviews of various methods. Some researchers have proposed that conduction/convection and evaporation are negligible in the absence of wind (e.g. Marciano and Harbeck, 1952), which is consistent with the assumption that only molecular processes contribute to the transfer of mass and heat without wind (Edinger et al 1974). Others have shown that significant conduction/convection and evaporation can occur in the absence of wind (e.g. Brady Graves and Geyer 1969, Harbeck 1962, Ryan and Harleman 1971, Helfrich et al 1982, and Adams et al 1987). This latter opinion has gained favor (Edinger et al, 1974), especially for waterbodies that exhibit water temperatures that are greater than the air temperature.

Brady, Graves, and Geyer (1969) pointed out that if the water surface temperature is warmer than the air temperature, then “the air adjacent to the water surface would tend to become both warmer and more moist than that above it, thereby (due to both of these factors) becoming less dense. The resulting vertical convective air currents … might be expected to achieve much higher rates of heat and mass transfer from the water surface [even in the absence of wind] than would be possible by molecular diffusion alone” (Edinger et al, 1974). Water temperatures in natural waterbodies are frequently greater than the air temperature, especially at night.

Edinger et al (1974) recommend that the relationship that was proposed by Brady, Graves and Geyer (1969) based on data from cooling ponds, could be representative of most environmental conditions. Shanahan (1984) recommends that the Lake Hefner equation (Marciano and Harbeck, 1952) is appropriate for natural waters in which the water temperature is less than the air temperature. Shanahan also recommends that the Ryan and Harleman (1971) equation as recalibrated by Helfrich et al (1982) is best suited for waterbodies that experience water temperatures that are greater than the air temperature. Adams et al (1987) revisited the Ryan and Harleman and Helfrich et al models and proposed another recalibration using additional data for waterbodies that exhibit water temperatures that are greater than the air temperature.

Three options are available on the **Light and Heat** worksheet in QUAL2K to calculate $f(U_w)$:

Brady, Graves, and Geyer (1969)

$$f(U_w) = 19.0 + 0.95U_w^2$$

where U_w = wind speed at a height of 7 m [m/s].

Adams et al. (1987)

Adams et al. (1987) updated the work of Ryan and Harleman (1971) and Helfrich et al (1982) to derive an empirical model of the wind speed function for heated waters that accounts for the enhancement of convection currents when the virtual temperature difference between the water and air ($\Delta\theta_v$ in degrees F) is greater than zero. Two wind functions reported by Adams et al., also known as the East Mesa method, are implemented in QUAL2K (wind speed in these equations is at a height of 2m).

- Adams 1: This formulation uses an empirical function to estimate effect of convection currents caused by virtual temperature differences between water and air, and the Harbeck (1962) equation is used to represent the contribution to conduction/convection and evaporation that is not due to convection currents caused by high virtual water temperature.

  $$f(U_w) = 0.271\sqrt{(22.4\Delta\theta_v^{1/3})^2 + (24.2A_{acres,i}^{-0.05}U_{w,mph})^2}$$

  where $U_{w,mph}$ is wind speed in mph and $A_{acres,i}$ is surface area of reach i in acres. The constant 0.271 converts the original units of BTU ft^{-2} day^{-1} mmHg^{-1} to cal cm^{-2} day^{-1} mmHg^{-1}.

- Adams 2: This formulation uses an empirical function of virtual temperature differences with the Marciano and Harbeck (1952) equation for the contribution to conduction/convection and evaporation that is not due to the high virtual water temperature

  $$f(U_w) = 0.271\sqrt{(22.4\Delta\theta_v^{1/3})^2 + (17U_{w,mph})^2}$$

Virtual temperature is defined as the temperature of dry air that has the same density as air under the in situ conditions of humidity. The virtual temperature difference between the water and air ($\Delta\theta_v$ in °F) accounts for the buoyancy of the moist air above a heated water surface. The virtual temperature difference is estimated from water temperature ($T_{w,f}$ in °F), air temperature ($T_{air,f}$ in °F), vapor pressure of water and air (e_s and e_{air} in mmHg), and the atmospheric pressure (p_{atm} is estimated as standard atmospheric pressure of 760 mmHg in QUAL2K):

$$\Delta\theta_v = \left(\frac{T_{w,f} + 460}{1 + 0.378e_s / p_{atm}} - 460\right) - \left(\frac{T_{air,f} + 460}{1 + 0.378e_{air} / p_{atm}} - 460\right) \tag{49}$$

The height of wind speed measurements is also an important consideration for estimating conduction/convection and evaporation. QUAL2K internally adjusts the wind speed to the correct height for the wind function that is selected on the **Light and Heat** worksheet. The input values for wind speed on the **Wind Speed** worksheet in QUAL2K are assumed to be representative of conditions at a height of 7 meters above the water surface. To convert wind speed measurements ($U_{w,z}$ in m/s) taken at any height (z_w in meters) to the equivalent conditions at a height of 7 m for input to the **Wind Speed** worksheet of QUAL2K, the exponential wind law equation may be used (TVA, 1972):

$$U_w = U_{wz}\left(\frac{z}{z_w}\right)^{0.15} \tag{50}$$

For example, if wind speed data were collected from a height of 2 m, then the wind speed at 7 m for input to the **Wind Speed** worksheet of QUAL2K would be estimated by multiplying the measured wind speed by a factor of 1.2.

4.1.5 Evaporation and Condensation

The heat loss due to evaporation can be represented by Dalton's law,

$$J_e = f(U_w)(e_s - e_{air}) \tag{51}$$

where e_s = the saturation vapor pressure at the water surface [mmHg], and e_{air} = the air vapor pressure [mmHg]. The saturation vapor pressure is computed as

$$e_{air} = 4.596e^{\frac{17.27T}{237.3+T}} \tag{52}$$

4.2 Sediment-Water Heat Transfer

A heat balance for bottom sediment underlying a water reach *i* can be written as

$$\frac{dT_{s,i}}{dt} = -\frac{J_{s,i}}{\rho_s C_{ps} H_{sed,i}} \tag{53}$$

where $T_{s,i}$ = the temperature of the bottom sediment below reach *i* [^{o}C], ρ_s = the density of the sediments [g/cm^3], C_{ps} = the specific heat of the sediments [cal/(g ^{o}C)], $J_{h,i}$ = the air-water heat flux [cal/(cm^2 d)], and $J_{s,i}$ = the sediment-water heat flux [cal/(cm^2 d)].
and $H_{sed,i}$ = the effective thickness of the sediment layer [cm].

The flux from the sediments to the water can be computed as

$$J_{s,i} = \rho_s C_{ps} \frac{\alpha_s}{H_{sed,i}/2}\left(T_{si} - T_i\right) \tag{54}$$

where α_s = the sediment thermal diffusivity [cm^2/s].

The thermal properties of some natural sediments along with its components are summarized in Table 4. Note that soft, gelatinous sediments found in the deposition zones of lakes are very porous and approach the values for water. Some very slow, impounded rivers may approach such a state. However, rivers will tend to have coarser sediments with significant fractions of sands, gravels and stones. Upland streams can have bottoms that are dominated by boulders and rock substrates.

Table 4 Thermal properties for natural sediments and the materials that comprise natural sediments.

Table 4. Thermal properties of various materials

Type of material	thermal conductivity		thermal diffusivity		ρ	C_p	ρCp	reference
	w/m/°C	cal/s/cm/°C	m^2/s	cm^2/s	g/cm3	cal/(g °C)	cal/(cm^3 °C)	
Sediment samples								
Mud Flat	1.82	0.0044	4.80E-07	0.0048			0.906	(1)
Sand	2.50	0.0060	7.90E-07	0.0079			0.757	"
Mud Sand	1.80	0.0043	5.10E-07	0.0051			0.844	"
Mud	1.70	0.0041	4.50E-07	0.0045			0.903	"
Wet Sand	1.67	0.0040	7.00E-07	0.0070			0.570	(2)
Sand 23% saturation with water	1.82	0.0044	1.26E-06	0.0126			0.345	(3)
Wet Peat	0.36	0.0009	1.20E-07	0.0012			0.717	(2)
Rock	1.76	0.0042	1.18E-06	0.0118			0.357	(4)
Loam 75% saturation with water	1.78	0.0043	6.00E-07	0.0060			0.709	(3)
Lake, gelatinous sediments	0.46	0.0011	2.00E-07	0.0020			0.550	(5)
Concrete canal	1.55	0.0037	8.00E-07	0.0080	2.200	0.210	0.460	"
Average of sediment samples:	*1.57*	*0.0037*	*6.45E-07*	*0.0064*			*0.647*	
Miscellaneous measurements:								
Lake, shoreline	0.59	0.0014						(5)
Lake soft sediments			3.25E-07	0.0033				"
Lake, with sand			4.00E-07	0.0040				"
River, sand bed			7.70E-07	0.0077				"
Component materials:								
Water	0.59	0.0014	1.40E-07	0.0014	1.000	0.999	1.000	(6)
Clay	1.30	0.0031	9.80E-07	0.0098	1.490	0.210	0.310	"
Soil, dry	1.09	0.0026	3.70E-07	0.0037	1.500	0.465	0.700	"
Sand	0.59	0.0014	4.70E-07	0.0047	1.520	0.190	0.290	"
Soil, wet	1.80	0.0043	4.50E-07	0.0045	1.810	0.525	0.950	"
Granite	2.89	0.0069	1.27E-06	0.0127	2.700	0.202	0.540	"
Average of composite materials:	*1.37*	*0.0033*	*6.13E-07*	*0.0061*	*1.670*	*0.432*	*0.632*	

(1) Andrews and Rodvey (1980)
(2) Geiger (1965)
(3) Nakshabandi and Kohnke (1965)
(4) Chow (1964) and Carslaw and Jaeger (1959)
(5) Hutchinson 1957, Jobson 1977, and Likens and Johnson 1969
(6) Cengel, Grigull, Mills, Bejan, Kreith and Bohn

Inspection of the component properties of Table 4 suggests that the presence of solid material in stream sediments leads to a higher coefficient of thermal diffusivity than that for water or porous lake sediments. In Q2K, we will use a value of 0.005 cm^2/s for this quantity.

In addition, specific heat tends to decrease with density. Thus, the product of these two quantities tends to be more constant than the multiplicands. Nevertheless, it appears that the presence of solid material in stream sediments leads to a higher product than that for water or gelatinous lake sediments. In Q2K, we will use a value of 0.7 cal/(cm^3 K) for this quantity. Finally, as derived in Appendix C, the sediment thickness is set to 10 cm in order to capture the effect of the sediments on the diel heat budget for the water.

5 CONSTITUENT MODEL

5.1 Constituents and General Mass Balance

The model constituents are listed in Table 5.

Table 5 Model state variables

Variable	Symbol	Units*
Conductivity	s	**μmhos**
Inorganic suspended solids	m_i	**mgD/L**
Dissolved oxygen	o	**mgO$_2$/L**
Slowly reacting CBOD	c_s	**mgO$_2$/L**
Fast reacting CBOD	c_f	**mgO$_2$/L**
Dissolved organic nitrogen	n_o	**μgN/L**
Ammonia nitrogen	n_a	**μgN/L**
Nitrate nitrogen	n_n	**μgN/L**
Dissolved organic phosphorus	p_o	**μgP/L**
Inorganic phosphorus	p_i	**μgP/L**
Phytoplankton	a_p	**μgA/L**
Detritus	m_o	**mgD/L**
Pathogen	x	**cfu/100 mL**
Alkalinity	Alk	**mgCaCO$_3$/L**
Total inorganic carbon	c_T	**mole/L**
Bottom algae	a_b	**gD/m^2**

* mg/L ≡ g/m^3

For all but the bottom algae, a general mass balance for a constituent in a reach is written as (Figure 15)

$$\frac{dc_i}{dt} = \frac{Q_{i-1}}{V_i} c_{i-1} - \frac{Q_i}{V_i} c_i - \frac{Q_{ab,i}}{V_i} c_i + \frac{E'_{i-1}}{V_i}(c_{i-1} - c_i) + \frac{E'_i}{V_i}(c_{i+1} - c_i) + \frac{W_i}{V_i} + S_i \qquad \textbf{(55)}$$

where W_i = the external loading of the constituent to reach i [g/d or mg/d], and S_i = sources and sinks of the constituent due to reactions and mass transfer mechanisms [g/m^3/d or mg/m^3/d].

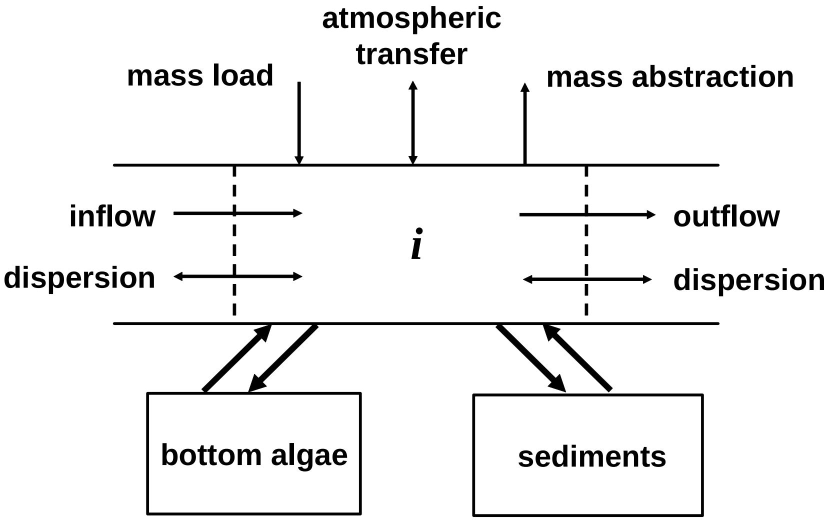


Figure 15 Mass balance.

The external load is computed as

$$W_i = \sum_{j=1}^{psi} Q_{ps,i,j} c_{psi,j} + \sum_{j=1}^{npsi} Q_{nps,i,j} c_{npsi,j} \tag{56}$$

where $c_{ps,i,j}$ is the jth point source concentration for reach i [mg/L or μg/L], and $c_{nps,i,j}$ is the jth non-point source concentration for reach i [mg/L or μg/L].

For bottom algae, the transport and loading terms are omitted,

$$\frac{da_{b,i}}{dt} = S_{b,i} \tag{57}$$

where $S_{b,i}$ = sources and sinks of bottom algae due to reactions [gD/m^2/d].

The sources and sinks for the state variables are depicted in Figure 16. The mathematical representation of these processes are presented in the following sections.

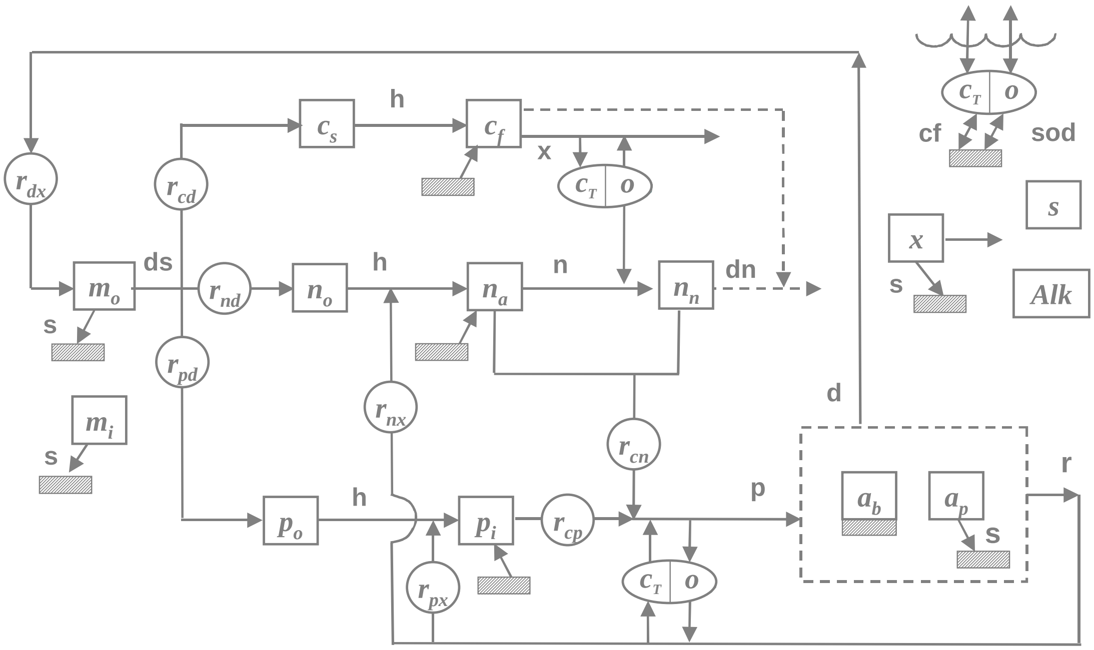


Figure 16 Model kinetics and mass transfer processes. The state variables are defined in Table 5. Kinetic processes are dissolution (ds), hydrolysis (h), oxidation (x), nitrification (n), denitrification (dn), photosynthesis (p), death (d), and respiration (r). Mass transfer processes are reaeration (re), settling (s), sediment oxygen demand (SOD), and sediment inorganic carbon flux (cf). Note that the subscript x for the stoichiometric conversions stands for chlorophyll *a* (*a*) and dry weight (*d*) for phytoplankton and bottom algae, respectively.

5.2 Reaction Fundamentals

5.2.1 Biochemical Reactions

The following chemical equations are used to represent the major biochemical reactions that take place in the model (Stumm and Morgan 1996):

Plant Photosynthesis and Respiration:

Ammonium as substrate:

$$106\mathrm{CO}_2 + 16\mathrm{NH}_4^+ + \mathrm{HPO}_4^{2-} + 108\mathrm{H}_2\mathrm{O} \underset{R}{\overset{P}{\rightleftarrows}} \mathrm{C}_{106}\mathrm{H}_{263}\mathrm{O}_{110}\mathrm{N}_{16}\mathrm{P}_1 + 107\mathrm{O}_2 + 14\mathrm{H}^+ \quad \textbf{(58)}$$

Nitrate as substrate:

$$106\mathrm{CO}_2 + 16\mathrm{NO}_3^- + \mathrm{HPO}_4^{2-} + 122\mathrm{H}_2\mathrm{O} + 18\mathrm{H}^+ \underset{R}{\overset{P}{\rightleftarrows}} \mathrm{C}_{106}\mathrm{H}_{263}\mathrm{O}_{110}\mathrm{N}_{16}\mathrm{P}_1 + 138\mathrm{O}_2 \quad \textbf{(59)}$$

Nitrification:

$$\mathrm{NH_4^+ + 2O_2 \rightarrow NO_3^- + H_2O + 2H^+} \tag{60}$$

Denitrification:

$$\mathrm{5CH_2O + 4NO_3^- + 4H^+ \rightarrow 5CO_2 + 2N_2 + 7H_2O} \tag{61}$$

Note that a number of additional reactions are used in the model such as those involved with simulating pH and unionized ammonia. These will be outlined when these topics are discussed later in this document.

5.2.2 Stoichiometry of Organic Matter

The model requires that the stoichiometry of organic matter (i.e., plants and detritus) be specified by the user. The following representation is suggested as a first approximation (Redfield et al. 1963, Chapra 1997),

$$100\,\mathrm{gD} : 40\,\mathrm{gC} : 7200\,\mathrm{mgN} : 1000\,\mathrm{mgP} : 1000\,\mathrm{mgA} \tag{62}$$

where gX = mass of element X [g] and mgY = mass of element Y [mg]. The terms D, C, N, P, and A refer to dry weight, carbon, nitrogen, phosphorus, and chlorophyll *a*, respectively. It should be noted that chlorophyll *a* is the most variable of these values with a range of approximately 500-2000 mgA (Laws and Chalup 1990, Chapra 1997).

These values are then combined to determine stoichiometric ratios as in

$$r_{xy} = \frac{\mathrm{gX}}{\mathrm{gY}} \tag{63}$$

For example, the amount of nitrogen that is released when 1 gD of detritus is dissolved can be computed as

$$r_{nd} = \frac{7200\,\mathrm{mgN}}{100\,\mathrm{gD}} = 72\,\frac{\mathrm{mgN}}{\mathrm{gD}}$$

5.2.2.1 Oxygen Generation and Consumption

The model requires that the rates of oxygen generation and consumption be prescribed. If ammonia is the substrate, the following ratio (based on Equation 62) can be used to determine the grams of oxygen generated for each gram of plant matter that is produced through photosynthesis.

$$r_{oca} = \frac{107\,\mathrm{moleO_2}(32\,\mathrm{gO_2/moleO_2})}{106\,\mathrm{moleC}(12\,\mathrm{gC/moleC})} = 2.69\,\frac{\mathrm{gO_2}}{\mathrm{gC}} \tag{64}$$

If nitrate is the substrate, the following ratio (based on Equation 63) applies

$$r_{ocn} = \frac{138\ \text{moleO}_2\,(32\ \text{gO}_2/\text{moleO}_2)}{106\ \text{moleC}(12\ \text{gC/moleC})} = 3.47\,\frac{\text{gO}_2}{\text{gC}} \tag{65}$$

Note that Equation (68) is also used for the stoichiometry of the amount of oxygen consumed for both plant respiration and fast organic CBOD oxidation.

For nitrification, the following ratio is based on Equation (64)

$$r_{on} = \frac{2\ \text{moleO}_2\,(32\ \text{gO}_2/\text{moleO}_2)}{1\ \text{moleN}(14\ \text{gN/moleN})} = 4.57\,\frac{\text{gO}_2}{\text{gN}} \tag{66}$$

5.2.2.2 CBOD Utilization Due to Denitrification

As represented by Equation (60), CBOD is utilized during denitrification,

$$r_{ondn} = 2.67\,\frac{\text{gO}_2}{\text{gC}}\,\frac{5\ \text{moleC}\times 12\ \text{gC/moleC}}{4\ \text{moleN}\times 14\ \text{gN/moleN}} \times \frac{1\ \text{gN}}{1000\ \text{mgN}} = 0.00286\,\frac{\text{gO}_2}{\text{mgN}} \tag{67}$$

5.2.3 Temperature Effects on Reactions

The temperature effect for all first-order reactions used in the model is represented by

$$k(T) = k(20)\theta^{T-20} \tag{68}$$

where $k(T)$ = the reaction rate [/d] at temperature T [^{o}C] and θ = the temperature coefficient for the reaction.

5.3 Constituent Reactions

The mathematical relationships that describe the individual reactions and concentrations of the model state variables (Table 5) are presented in the following paragraphs.

5.3.1 Conservative Substance (*s*)

By definition, conservative substances are not subject to reactions:

$$S_s = 0 \tag{69}$$

5.3.2 Phytoplankton (a_p)

Phytoplankton increase due to photosynthesis. They are lost via respiration, death, and settling

$$S_{ap} = \text{PhytoPhoto} - \text{PhytoResp} - \text{PhytoDeath} - \text{PhytoSettl} \tag{70}$$

5.3.2.1 Photosynthesis

Phytoplankton photosynthesis is a function of temperature, nutrients, and light

$$\text{PhytoPhoto} = \mu_p a_p \tag{71}$$

where μ_p = phytoplankton photosynthesis rate [/d], which is calculated as

$$\mu_p = k_{gp}(T)\phi_{Np}\phi_{Lp} \tag{72}$$

where $k_{gp}(T)$ = the maximum photosynthesis rate at temperature T [/d], ϕ_{Np} = phytoplankton nutrient attenuation factor [dimensionless number between 0 and 1], and ϕ_{Lp} = the phytoplankton light attenuation coefficient [dimensionless number between 0 and 1].

Nutrient Limitation. Michaelis-Menten equations are used to represent growth limitation for inorganic nitrogen and phosphorus. The minimum value is then used to compute the nutrient attenuation factor,

$$\phi_{Np} = \min\left(\frac{n_a + n_n}{k_{sNp} + n_a + n_n}, \frac{p_i}{k_{sPp} + p_i}\right) \tag{73}$$

where k_{sNp} = nitrogen half-saturation constant [μgN/L] and k_{sPp} = phosphorus half-saturation constant [μgP/L].

Light Limitation. It is assumed that light attenuation through the water follows the Beer-Lambert law,

$$PAR(z) = PAR(0)e^{-k_e z} \tag{74}$$

where $PAR(z)$ = photosynthetically available radiation (PAR) at depth z below the water surface [ly/d][2], and k_e = the light extinction coefficient [m^{-1}]. The PAR at the water surface is assumed to be a fixed fraction of the solar radiation

$$PAR(0) = 0.47\, I(0) \tag{75}$$

The extinction coefficient is related to model variables by

$$k_e = k_{eb} + \alpha_i m_i + \alpha_o m_o + \alpha_p a_p + \alpha_{pn} a_p^{2/3} \tag{76}$$

where k_{eb} = the background coefficient accounting for extinction due to water and color [/m], α_i, α_o, α_p, and α_{pn}, are constants accounting for the impacts of inorganic suspended solids [L/mgD/m], particulate organic matter [L/mgD/m], and chlorophyll [L/μgA/m and $(\text{L}/\mu\text{gA})^{2/3}$/m], respectively. Suggested values for these coefficients are listed in Table 6.

---

[2] ly/d = langley per day. A langley is equal to a calorie per square centimeter. Note that a ly/d is related to the $\mu\text{E/m}^2$/d by the following approximation: 1 $\mu\text{E/m}^2/\text{s} \cong 0.45$ Langley/day (LIC-OR, Lincoln, NE).

Table 6 Suggested values for light extinction coefficients

Symbol	Value	Reference
α_i	0.052	Di Toro (1978)
α_o	0.174	Di Toro (1978)
α_p	0.0088	Riley (1956)
α_{pn}	0.054	Riley (1956)

Three models are used to characterize the impact of light on phytoplankton photosynthesis (Figure 17):

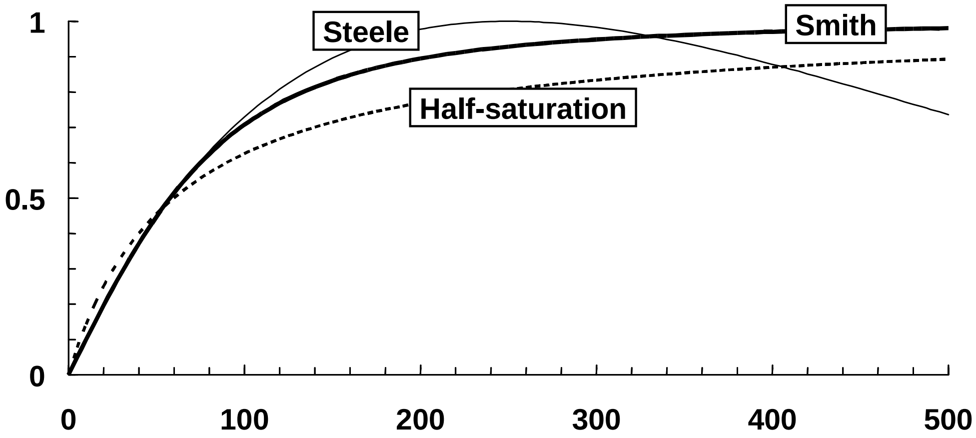


Figure 17 The three models used for phytoplankton and bottom algae photosynthetic light dependence. The plot shows growth attenuation versus PAR intensity [ly/d].

<u>Half-Saturation (Michaelis-Menten) Light Model:</u>

$$F_{Lp} = \frac{I(z)}{K_{Lp} + I(z)} \tag{77}$$

where F_{Lp} = phytoplankton growth attenuation due to light and K_{Lp} = the phytoplankton light parameter. In the case of the half-saturation model, the light parameter is a half-saturation coefficient [ly/d]. This function can be combined with the Beer-Lambert law and integrated over water depth, H [m], to yield the phytoplankton light attenuation coefficient

$$\phi_{Lp} = \frac{1}{k_e H} \ln\left(\frac{K_{Lp} + I(0)}{K_{Lp} + I(0)e^{-k_e H}}\right) \tag{78}$$

<u>Smith's Function:</u>

$$F_{Lp} = \frac{I(z)}{\sqrt{K_{Lp}^2 + I(z)^2}} \tag{79}$$

where K_{Lp} = the Smith parameter for phytoplankton [ly/d]; that is, the PAR at which growth is 70.7% of the maximum. This function can be combined with the Beer-Lambert law and integrated over water depth to yield

$$\phi_{Lp} = \frac{1}{k_e H} \ln\left(\frac{I(0)/K_{Lp} + \sqrt{1 + \left(I(0)/K_{Lp}\right)^2}}{\left(I(0)/K_{Lp}\right)e^{-k_e H} + \sqrt{1 + \left(\left(I(0)/K_{Lp}\right)e^{-k_e H}\right)^2}} \right) \tag{80}$$

Steele's Equation:

$$F_{Lp} = \frac{I(z)}{K_{Lp}} e^{1 - \frac{I(z)}{K_{Lp}}} \tag{81}$$

where K_{Lp} = the PAR at which phytoplankton growth is optimal [ly/d]. This function can be combined with the Beer-Lambert law and integrated over water depth to yield

$$\phi_{Lp} = \frac{2.718282}{k_e H} \left(e^{-\frac{I(0)}{K_{Lp}} e^{-k_e H}} - e^{-\frac{I(0)}{K_{Lp}}} \right) \tag{82}$$

5.3.2.2 Losses

Respiration. Phytoplankton respiration is represented as a first-order rate that is attenuated at low oxygen concentration,

$$\text{PhytoResp} = k_{rp}(T)\, a_p \tag{83}$$

where $k_{rp}(T)$ = temperature-dependent phytoplankton respiration rate [/d].

Death. Phytoplankton death is represented as a first-order rate,

$$\text{PhytoDeath} = k_{dp}(T) a_p \tag{84}$$

where $k_{dp}(T)$ = temperature-dependent phytoplankton death rate [/d].

Settling. Phytoplankton settling is represented as

$$\text{PhytoSettl} = \frac{v_a}{H} a_p \tag{85}$$

where v_a = phytoplankton settling velocity [m/d].

5.3.3 Bottom algae (a_b)

Bottom algae increase due to photosynthesis. They are lost via respiration and death.

$$S_{ab} = \text{BotAlgPhoto} - \text{BotAlgResp} - \text{BotAlgDeath} \tag{86}$$

5.3.3.1 Photosynthesis

The representation of bottom algae photosynthesis is a simplification of a model developed by Rutherford et al. (1999). Photosynthesis is based on a temperature-corrected zero-order rate attenuated by nutrient and light limitation,

$$\text{BotAlgPhoto} = C_{gb}(T)\phi_{Nb}\phi_{Lb} \tag{87}$$

where $C_{gb}(T)$ = the temperature-dependent maximum photosynthesis rate [gD/(m^2 d)], ϕ_{Nb} = bottom algae nutrient attenuation factor [dimensionless number between 0 and 1], and ϕ_{Lb} = the bottom algae light attenuation coefficient [dimensionless number between 0 and 1].

Temperature Effect. As for the first-order rates, an Arrhenius model is employed to quantify the effect of temperature on bottom algae photosynthesis,

$$C_{gb}(T) = C_{gb}(20)\theta^{T-20} \tag{88}$$

Nutrient Limitation. Michaelis-Menten equations are used to represent growth limitation due to inorganic nitrogen and phosphorus. The minimum value is then used to compute the nutrient attenuation coefficient,

$$\phi_{Nb} = \min\left(\frac{n_a + n_n}{k_{sNb} + n_a + n_n}, \frac{p_i}{k_{sPb} + p_i}\right) \tag{89}$$

where k_{sNb} = nitrogen half-saturation constant [µgN/L] and k_{sPb} = phosphorus half-saturation constant [µgP/L].

Light Limitation. In contrast to the phytoplankton, light limitation at any time is determined by the amount of PAR reaching the bottom of the water column. This quantity is computed with the Beer-Lambert law (recall Equation 78) evaluated at the bottom of the river,

$$I(H) = I(0)e^{-k_e H} \tag{90}$$

As with the phytoplankton, three models (Equations 81, 83, and 85) are used to characterize the impact of light on bottom algae photosynthesis. Substituting Equation (97) into these models yields the following formulas for the bottom algae light attenuation coefficient,

<u>Half-Saturation Light Model:</u>

$$\phi_{Lb} = \frac{I(0)e^{-k_e H}}{K_{Lb} + I(0)e^{-k_e H}} \tag{91}$$

<u>Smith's Function:</u>

$$\phi_{Lp} = \frac{I(0)e^{-k_e H}}{\sqrt{K_{Lb}^2 + \left(I(0)e^{-k_e H}\right)^2}} \tag{92}$$

Steele's Equation:

$$\phi_{Lb} = \frac{I(0)e^{-k_e H}}{K_{Lb}} e^{1+\frac{I(0)e^{-k_e H}}{K_{Lb}}} \tag{93}$$

where K_{Lb} = the appropriate bottom algae light parameter for each light model.

5.3.3.2 Losses

Respiration. Bottom algae respiration is represented as a first-order rate that is attenuated at low oxygen concentration,

$$\text{BotAlgResp} = k_{rb}(T)\, a_b \tag{94}$$

where $k_{rb}(T)$ = temperature-dependent bottom algae respiration rate [/d].

Death. Bottom algae death is represented as a first-order rate,

$$\text{BotAlgDeath} = k_{db}(T) a_b \tag{95}$$

where $k_{db}(T)$ = the temperature-dependent bottom algae death rate [/d].

5.3.4 Detritus (m_o)

Detritus or particulate organic matter (POM) increases due to plant death. It is lost via dissolution and settling

$$S_{mo} = r_{da}\ \text{PhytoDeath} + \text{BotAlgDeath} - \text{DetrDiss} - \text{DetrSettl} \tag{96}$$

where

$$\text{DetrDiss} = k_{dt}(T) m_o \tag{97}$$

where $k_{dt}(T)$ = the temperature-dependent detritus dissolution rate [/d] and

$$\text{DetrSettl} = \frac{v_{dt}}{H} m_o \tag{98}$$

where v_{dt} = detritus settling velocity [m/d].

5.3.5 Slowly Reacting CBOD (c_s)

Slowly reacting CBOD increases due to detritus dissolution. It is lost via hydrolysis.

$$S_{cs} = r_{od}\ \text{DetrDiss} - \text{SlowCHydr} \tag{99}$$

where

$$\text{SlowCHydr} = k_{hc}(T)c_s \tag{100}$$

where $k_{hc}(T)$ = the temperature-dependent slow CBOD hydrolysis rate [/d].

5.3.6 Fast Reacting CBOD (c_f)

Fast reacting CBOD is gained via the hydrolysis of slowly-reacting CBOD. It is lost via oxidation and denitrification.

$$S_{cf} = \text{SlowCHydr} - \text{FastCOxid} - r_{ondn}\text{Denitr} \tag{101}$$

where

$$\text{FastCOxid} = F_{oxcf}\ k_{dc}(T)c_f \tag{102}$$

where $k_{dc}(T)$ = the temperature-dependent fast CBOD oxidation rate [/d] and F_{oxcf} = attenuation due to low oxygen [dimensionless]. The parameter r_{ondn} is the ratio of oxygen equivalents lost per nitrate nitrogen that is denitrified (Equation 71). The term Denitr is the rate of denitrification [μgN/L/d]. It will be defined in 5.3.10 below.

Three formulations are used to represent the oxygen attenuation:

Half-Saturation:

$$F_{oxrp} = \frac{o}{K_{socf} + o} \tag{103}$$

where K_{socf} = half-saturation constant for the effect of oxygen on fast CBOD oxidation [mgO$_2$/L].

Exponential:

$$F_{oxrp} = (1 - e^{-K_{socf}o}) \tag{104}$$

where K_{socf} = exponential coefficient for the effect of oxygen on fast CBOD oxidation [L/mgO$_2$].

Second-Order Half Saturation:

$$F_{oxrp} = \frac{o^2}{K_{socf} + o^2} \tag{105}$$

where K_{socf} = half-saturation constant for second-order effect of oxygen on fast CBOD oxidation [$\text{mgO}_2^2/\text{L}^2$].

5.3.7 Dissolved Organic Nitrogen (n_o)

Dissolved organic nitrogen increases due to detritus dissolution. It is lost via hydrolysis.

$$S_{no} = r_{nd}\ \text{DetrDiss} - \text{DONHydr} \tag{106}$$

$$\text{DONHydr} = k_{hn}(T)n_o \tag{107}$$

where $k_{hn}(T)$ = the temperature-dependent organic nitrogen hydrolysis rate [/d].

5.3.8 Ammonia Nitrogen (n_a)

Ammonia nitrogen increases due to dissolved organic nitrogen hydrolysis and plant respiration. It is lost via nitrification and plant photosynthesis:

$$\begin{aligned} S_{na} = \text{DONHydr} + r_{na}\ \text{PhytoResp} + r_{nd}\ \text{BotAlgResp} - \text{NH4Nitrif} \\ - r_{na}P_{ap}\ \text{PhytoPhoto} - r_{nd}\ P_{ab}\ \text{BotAlgPhoto} \end{aligned} \tag{108}$$

The ammonia nitrification rate is computed as

$$\text{NH4Nitrif} = F_{oxna}k_n(T)n_a \tag{109}$$

where $k_n(T)$ = the temperature-dependent nitrification rate for ammonia nitrogen [/d] and F_{oxna} = attenuation due to low oxygen [dimensionless]. Oxygen attenuation is modeled by Equations (106) to (108) with the oxygen dependency represented by the parameter K_{sona}.

The coefficients P_{ap} and P_{ap} are the preferences for ammonium as a nitrogen source for phytoplankton and bottom algae, respectively,

$$P_{ap} = \frac{n_a n_n}{(k_{hnxp} + n_a)(k_{hnxp} + n_n)} + \frac{n_a k_{hnxp}}{(n_a + n_n)(k_{hnxp} + n_n)} \tag{110}$$

$$P_{ab} = \frac{n_a n_n}{(k_{hnxb} + n_a)(k_{hnxb} + n_n)} + \frac{n_a k_{hnxb}}{(n_a + n_n)(k_{hnxb} + n_n)} \tag{111}$$

where k_{hnxp} = preference coefficient of phytoplankton for ammonium [mgN/m^3] and k_{hnxb} = preference coefficient of bottom algae for ammonium [mgN/m^3].

5.3.9 Unionized Ammonia

The model simulates total ammonia. In water, the total ammonia consists of two forms: ammonium ion, NH_4^+, and unionized ammonia, NH_3. At normal pH (6 to 8), most of the total ammonia will be in the ionic form. However at high pH, unionized ammonia predominates. The amount of unionized ammonia can be computed as

$$n_{au} = F_u n_a \tag{112}$$

where n_{au} = the concentration of unionized ammonia [μgN/L], and F_u = the fraction of the total ammonia that is in unionized form,

$$F_u = \frac{1}{1 + 10^{-\mathrm{pH}} / K_a} \tag{113}$$

where K_a = the equilibrium coefficient for the ammonia dissociation reaction, which is related to temperature by

$$\mathrm{p}K_a = 0.09018 + \frac{2729.92}{T_a} \tag{114}$$

where T_a is absolute temperature [K] and $\mathrm{p}K_a = -\log_{10}(K_a)$.

5.3.10 Nitrate Nitrogen (n_n)

Nitrate nitrogen increases due to nitrification of ammonia. It is lost via denitrification and plant photosynthesis:

$$\begin{aligned} S_{ni} = \mathrm{NH4Nitrif} - \mathrm{Denitr} - r_{na}(1 - P_{ap})\,\mathrm{PhytoPhoto} \\ - r_{nd}(1 - P_{ab})\,\mathrm{BotAlgPhoto} \end{aligned} \tag{115}$$

The denitrification rate is computed as

$$\mathrm{Denitr} = (1 - F_{oxdn}) k_{dn}(T) n_n \tag{116}$$

where $k_{dn}(T)$ = the temperature-dependent denitrification rate of nitrate nitrogen [/d] and F_{oxdn} = effect of low oxygen on denitrification [dimensionless] as modeled by Equations (106) to (108) with the oxygen dependency represented by the parameter K_{sodn}.

5.3.11 Dissolved Organic Phosphorus (p_o)

Dissolved organic phosphorus increases due to dissolution of detritus. It is lost via hydrolysis.

$$S_{po} = r_{pd}\,\mathrm{DetrDiss} - \mathrm{DOPHydr} \tag{117}$$

where

$$\text{DOPHydr} = k_{hp}(T)\,p_o \tag{118}$$

where $k_{hp}(T)$ = the temperature-dependent organic phosphorus hydrolysis rate [/d].

5.3.12 Inorganic Phosphorus (p_i)

Inorganic phosphorus increases due to dissolved organic phosphorus hydrolysis and plant respiration. It is lost via plant photosynthesis:

$$\begin{aligned} S_{pi} = {} & \text{DOPHydr} + r_{pa}\text{PhytoResp} + r_{pd}\ \text{BotAlgResp} \\ & - r_{pa}\text{PhytoPhoto} - r_{pd}\text{BotAlgPhoto} \end{aligned} \tag{119}$$

5.3.13 Inorganic Suspended Solids (m_i)

Inorganic suspended solids are lost via settling,

$$S_{mi} = -\ \text{InorgSettl}$$

where

$$\text{InorgSettl} = \frac{v_i}{H} m_i \tag{120}$$

where v_i = inorganic suspended solids settling velocity [m/d].

5.3.14 Dissolved Oxygen (o)

Dissolved oxygen increases due to plant photosynthesis. It is lost via fast CBOD oxidation, nitrification and plant respiration. Depending on whether the water is undersaturated or oversaturated it is gained or lost via reaeration,

$$\begin{aligned} S_o = {} & r_{oa}\,\text{PhytoGrowth} + r_{od}\text{BotAlgGrowth} - r_{oc}\text{FastCOxid} - r_{on}\text{NH4Nitr} \\ & - r_{oa}\text{PhytoResp} - r_{od}\text{BotAlgResp} + \text{OxReaer} \end{aligned} \tag{121}$$

where

$$\text{OxReaer} = k_a(T)\big(o_s(T, elev) - o\big) \tag{122}$$

where $k_a(T)$ = the temperature-dependent oxygen reaeration coefficient [/d], $o_s(T, elev)$ = the saturation concentration of oxygen [mgO$_2$/L] at temperature, T, and elevation above sea level, $elev$.

5.3.14.1 Oxygen Saturation

The following equation is used to represent the dependence of oxygen saturation on temperature (APHA 1992)

$$\ln o_s(T,0) = -139.34411 + \frac{1.575701\times 10^5}{T_a} - \frac{6.642308\times 10^7}{T_a^2} + \frac{1.243800\times 10^{10}}{T_a^3} - \frac{8.621949\times 10^{11}}{T_a^4} \tag{123}$$

where $o_s(T, 0)$ = the saturation concentration of dissolved oxygen in freshwater at 1 atm [mgO$_2$/L] and T_a = absolute temperature [K] where $T_a = T + 273.15$.

The effect of elevation is accounted for by

$$o_s(T, elev) = e^{\ln o_s(T,0)}(1 - 0.0001148 elev) \tag{124}$$

where *elev* = the elevation above sea level [m].

5.3.14.2 Reaeration Formulas

The reaeration coefficient can be prescribed on the **Reach** worksheet. If reaeration is not prescribed, it can be computed using one of the following formulas:

O'Connor-Dobbins:

$$k_a(20) = 3.93\frac{U^{0.5}}{H^{1.5}} \tag{125}$$

Owens-Gibbs:

$$k_a(20) = 5.32\frac{U^{0.67}}{H^{1.85}} \tag{126}$$

Churchill:

$$k_a(20) = 5.026\frac{U}{H^{1.67}} \tag{127}$$

where U = velocity [m/s] and H = depth [m].

Reaeration can also be internally calculated based on the following scheme patterned after a plot developed by Covar (1976) (Figure 18):

- ☐ If $H < 0.61$ m, use the Owens-Gibbs formula

☐ If $H > 0.61$ m and $H > 3.45U^{2.5}$, use the O'Connor-Dobbins formula
☐ Otherwise, use the Churchill formula

This is referred to as option **Internal** on the **Rates** worksheet of Q2K.

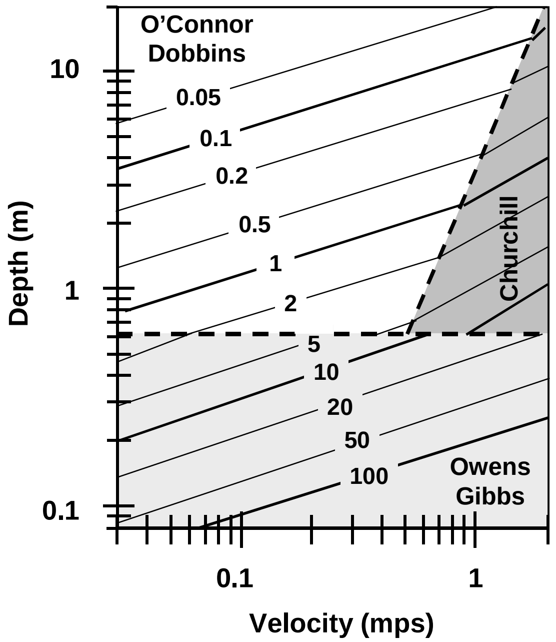


Figure 18 Reaeration rate (/d) versus depth and velocity (Covar 1976).

5.3.14.3 Effect of Control Structures: Oxygen

Oxygen transfer in streams is influenced by the presence of control structures such as weirs, dams, locks, and waterfalls (Figure 19). Butts and Evans (1983) have reviewed efforts to characterize this transfer and have suggested the following formula,

$$r_d = 1 + 0.38 a_d b_d H_d (1 - 0.11 H_d)(1 + 0.046 T) \tag{128}$$

where r_d = the ratio of the deficit above and below the dam, H_d = the difference in water elevation [m] as calculated with Equation (7), T = water temperature (°C) and a_d and b_d are coefficients that correct for water-quality and dam-type. Values of these coefficients are summarized in Table 7.

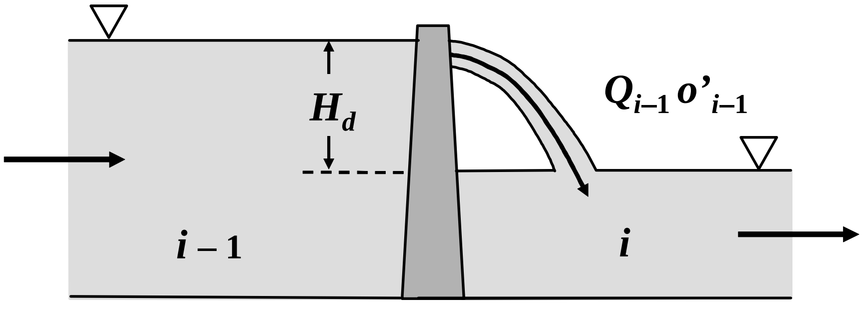


Figure 19 Water flowing over a river control structure.

Table 7 Coefficient values used to predict the effect of dams on stream reaeration.

(*a*) Water-quality coefficient	
Polluted state	a_d
Gross	0.65
Moderate	1.0
Slight	1.6
Clean	1.8
(*b*) Dam-type coefficient	
Dam type	b_d
Flat broad-crested regular step	0.70
Flat broad-crested irregular step	0.80
Flat broad-crested vertical face	0.60
Flat broad-crested straight-slope face	0.75
Flat broad-crested curved face	0.45
Round broad-crested curved face	0.75
Sharp-crested straight-slope face	1.00
Sharp-crested vertical face	0.80
Sluice gates	0.05

The oxygen mass balance for the reach below the structure is written as

$$\frac{do_i}{dt} = \frac{Q_{i-1}}{V_i} o'_{i-1} - \frac{Q_i}{V_i} o_i - \frac{Q_{ab,i}}{V_i} o_i + \frac{E'_i}{V_i}\left(o_{i+1} - o_i\right) + \frac{W_{o,i}}{V_i} + S_{o,i} \tag{129}$$

where o'_{i-1} = the oxygen concentration entering the reach [mgO$_2$/L], where

$$o'_{i-1} = o_{s,i-1} - \frac{o_{s,i-1} - o_{i-1}}{r_d} \tag{130}$$

5.3.15 Pathogen (*x*)

Pathogens are subject to death and settling,

$$S_x = -\text{PathDeath} - \text{PathSettl} \tag{131}$$

5.3.15.1 Death

Pathogen death is due to natural die-off and light (Chapra 1997). The death of pathogens in the absence of light is modeled as a first-order temperature-dependent decay and the death rate due to light is based on the Beer-Lambert law,

$$\text{PathDeath} = k_{dx}(T)x + \frac{I(0)/24}{k_e H}\left(1 - e^{-k_e H}\right)x \tag{132}$$

where $k_{dx}(T)$ = temperature-dependent pathogen die-off rate [/d].

5.3.15.2 Settling

Pathogen settling is represented as

$$\text{PathSettl} = \frac{v_x}{H}x \tag{133}$$

where v_x = pathogen settling velocity [m/d].

5.3.16 pH

The following equilibrium, mass balance and electroneutrality equations define a freshwater dominated by inorganic carbon (Stumm and Morgan 1996),

$$K_1 = \frac{[\mathrm{HCO_3^-}][\mathrm{H^+}]}{[\mathrm{H_2CO_3^*}]} \tag{134}$$

$$K_2 = \frac{[\mathrm{CO_3^{2-}}][\mathrm{H^+}]}{[\mathrm{HCO_3^-}]} \tag{135}$$

$$K_w = [\mathrm{H^+}][\mathrm{OH^-}] \tag{136}$$

$$c_T = [\mathrm{H_2CO_3^*}] + [\mathrm{HCO_3^-}] + [\mathrm{CO_3^{2-}}] \tag{137}$$

$$Alk = [\mathrm{HCO_3^-}] + 2[\mathrm{CO_3^{2-}}] + [\mathrm{OH^-}] - [\mathrm{H^+}] \tag{138}$$

where K_1, K_2 and K_w are acidity constants, Alk = alkalinity [eq L^{-1}], H_2CO_3* = the sum of dissolved carbon dioxide and carbonic acid, HCO_3^- = bicarbonate ion, CO_3^{2-} = carbonate ion, H^+ = hydronium ion, OH^- = hydroxyl ion, and c_T = total inorganic carbon concentration [mole L^{-1}]. The brackets [ ] designate molar concentrations.

Note that the alkalinity is expressed in units of eq/L for the internal calculations. For input and output, it is expressed as mgCaCO$_3$/L. The two units are related by

$$Alk(\mathrm{mgCaCO_3/L}) = 50{,}000 \times Alk(\mathrm{eq/L}) \tag{139}$$

The equilibrium constants are corrected for temperature by

Harned and Hamer (1933):

$$\mathrm{p}K_w = \frac{4787.3}{T_a} + 7.1321\log_{10}(T_a) + 0.010365T_a - 22.80 \tag{140}$$

Plummer and Busenberg (1982):

$$\log K_1 = -356.3094 - 0.06091964T_a + 21834.37/T_a \\ + 126.8339\log T_a - 1{,}684{,}915/T_a^2 \tag{141}$$

Plummer and Busenberg (1982):

$$\log K_2 = -107.8871 - 0.03252849T_a + 5151.79/T_a \\ + 38.92561\log T_a - 563{,}713.9/T_a^2 \tag{142}$$

The nonlinear system of five simultaneous equations (138 through 142) can be solved numerically for the five unknowns: $[H_2CO_3{}^*]$, $[HCO_3{}^-]$, $[CO_3{}^{2-}]$, $[OH^-]$, and $\{H^+\}$. As presented Stumm and Morgan (1996), an efficient solution method can be derived by combining Equations (138), (139) and (141) to define the quantities (Stumm and Morgan 1996)

$$\alpha_0 = \frac{[\mathrm{H}^+]^2}{[\mathrm{H}^+]^2 + K_1[\mathrm{H}^+] + K_1K_2} \tag{143}$$

$$\alpha_1 = \frac{K_1[\mathrm{H}^+]}{[\mathrm{H}^+]^2 + K_1[\mathrm{H}^+] + K_1K_2} \tag{144}$$

$$\alpha_2 = \frac{K_1K_2}{[\mathrm{H}^+]^2 + K_1[\mathrm{H}^+] + K_1K_2} \tag{145}$$

where α_0, α_1, and α_2 = the fraction of total inorganic carbon in carbon dioxide, bicarbonate, and carbonate, respectively. Equations (140), (142), (148) and (149) can then be combined to yield,

$$\mathrm{Alk} = (\alpha_1 + 2\alpha_2)c_T + \frac{K_w}{[\mathrm{H}^+]} - [\mathrm{H}^+] \tag{146}$$

Thus, solving for pH reduces to determining the root, $\{H^+\}$, of

$$f([\mathrm{H}^+]) = (\alpha_1 + 2\alpha_2)c_T + \frac{K_w}{[\mathrm{H}^+]} - [\mathrm{H}^+] - \mathrm{Alk} \tag{147}$$

where pH is then calculated with

$$\mathrm{pH} = -\log_{10}[\mathrm{H}^+] \tag{148}$$

The root of Equation (151) is determined with the bisection method (Chapra and Canale 2002).

5.3.17 Total Inorganic Carbon (c_T)

Total inorganic carbon concentration increases due to fast carbon oxidation and plant respiration. It is lost via plant photosynthesis. Depending on whether the water is undersaturated or oversaturated with CO_2, it is gained or lost via reaeration,

$$\begin{aligned} S_{cT} = r_{ccc}\,\text{FastCOxid} + r_{cca}\,\text{PhytoResp} + r_{ccd}\,\text{BotAlgResp} \\ - r_{cca}\,\text{PhytoPhoto} - r_{ccd}\,\text{BotAlgPhoto} + \text{CO2Reaer} \end{aligned} \tag{149}$$

where

$$\text{CO2Reaer} = k_{ac}(T)\left([\text{CO}_2]_s - \alpha_0 c_T\right) \tag{150}$$

where $k_{ac}(T)$ = the temperature-dependent carbon dioxide reaeration coefficient [/d], and $[CO2]_s$ = the saturation concentration of carbon dioxide [mole/L].

The stoichiometric coefficients are derived from Equation (62)[3]

$$r_{cca} = r_{ca}\left(\frac{\text{gC}}{\text{mgA}}\right) \times \frac{\text{moleC}}{12\text{ gC}} \times \frac{\text{m}^3}{1000\text{ L}} \tag{151}$$

$$r_{ccd} = r_{cd}\left(\frac{\text{gC}}{\text{gD}}\right) \times \frac{\text{moleC}}{12\text{ gC}} \times \frac{\text{m}^3}{1000\text{ L}} \tag{152}$$

$$r_{ccc} = \frac{\text{moleC}}{12\text{ gC}} \times \frac{\text{m}^3}{1000\text{ L}} \tag{153}$$

5.3.17.1 Carbon Dioxide Saturation

The CO_2 saturation is computed with Henry's law,

$$[\text{CO}_2]_\text{s} = K_H \text{p}_{\text{CO}_2} \tag{154}$$

where K_H = Henry's constant [mole (L atm)$^{-1}$] and p_{CO_2} = the partial pressure of carbon dioxide in the atmosphere [atm]. Note that users input the partial pressure to Q2K in ppm. The program internally converts ppm to atm using the conversion: 10^{-6} atm/ppm.

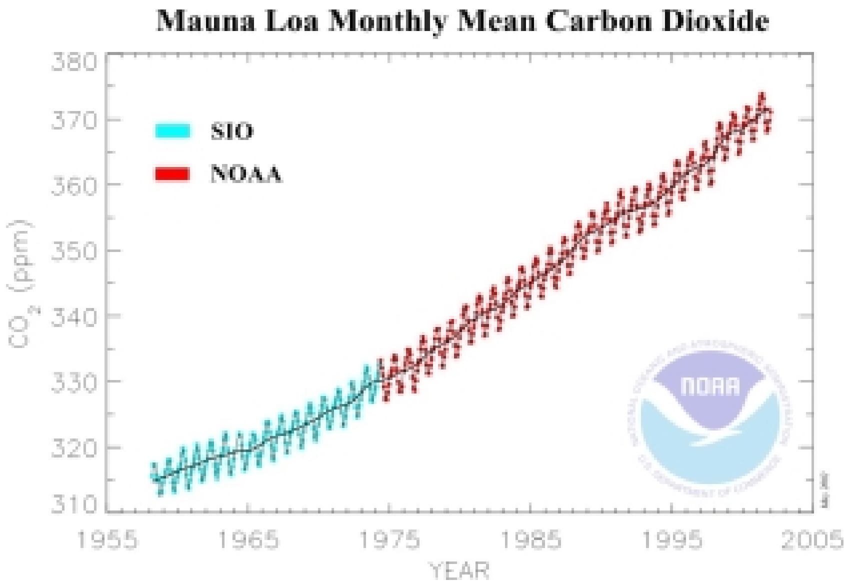
The value of K_H can be computed as a function of temperature by (Edmond and Gieskes 1970)

$$\text{p}K_\text{H} = -\frac{2385.73}{T_a} - 0.0152642 T_a + 14.0184 \tag{155}$$

---

[3] The conversion, m^3 = 1000 L is included because all mass balances express volume in m^3, whereas total inorganic carbon is expressed as mole/L.

The partial pressure of CO_2 in the atmosphere has been increasing, largely due to the combustion of fossil fuels (Figure 20). Values in 2003 are approximately $10^{-3.43}$ atm (= 372 ppm).



Atmospheric carbon dioxide monthly mean mixing ratios. Data prior to May 1974 are from the Scripps Institution of Oceanography (SIO, blue), data since May 1974 are from the National Oceanic and Atmospheric Administration (NOAA, red). A long-term trend curve is fitted to the monthly mean values. Principal investigators: Dr. Pieter Tans, NOAA CMDL Carbon Cycle Greenhouse Gases, Boulder, Colorado, (303) 497-6678, ptans@cmdl.noaa.gov, and Dr. Charles D. Keeling, SIO, La Jolla, California, (858) 534-6001, cdkeeling@ucsd.edu.

Figure 20 Concentration of carbon dioxide in the atmosphere as recorded at Mauna Loa Observatory, Hawaii. (http://www.cmdl.noaa.gov/ccg/figures/co2mm_mlo.jpg).

5.3.17.2 CO_2 Gas Transfer

The CO_2 reaeration coefficient can be computed from the oxygen reaeration rate by

$$k_{ac}(20) = \left(\frac{32}{44}\right)^{0.25} k_a(20) = 0.923\, k_a(20) \qquad \textbf{(156)}$$

5.3.17.3 Effect of Control Structures: CO_2

As was the case for dissolved oxygen, carbon dioxide gas transfer in streams can be influenced by the presence of control structures. Q2K assumes that carbon dioxide behaves similarly to dissolved oxygen (recall Sec. 5.3.14.3). Thus, the inorganic carbon mass balance for the reach immediately downstream of the structure is written as

$$\frac{dc_{T,i}}{dt} = \frac{Q_{i-1}}{V_i} c'_{T,i-1} - \frac{Q_i}{V_i} c_{T,i} - \frac{Q_{ab,i}}{V_i} c_{T,i} + \frac{E_i'}{V_i}\left(c_{T,i+1} - c_{T,i}\right) + \frac{W_{cT,i}}{V_i} + S_{cT,i} \qquad \textbf{(157)}$$

where $c'_{T,i-1}$ = the concentration of inorganic carbon entering the reach [mgO_2/L], where

$$c'_{T,i-1} = (\alpha_1 + \alpha_2)c_{T,i-1} + CO_{2,s,i-1} - \frac{CO_{2,s,i-1} - \alpha_2 c_{T,i-1}}{r_d} \tag{158}$$

where r_d is calculated with Equation (132).

5.3.18 Alkalinity (Alk)

The present model accounts for changes in alkalinity due to plant photosynthesis and respiration, nitrification, and denitrification.

$$\begin{aligned} S_{alk} &= \left(-r_{alkaa}P_{ap} + r_{alkan}(1-P_{ap})\right)\text{PhytoPhoto} + r_{alkaa}\text{PhytoResp} \\ &\quad + \left(-r_{alkda}P_{ap} + r_{alkdn}(1-P_{ap})\right)\text{BotAlgGrowth} + r_{alkda}\text{BotAlgResp} \\ &\quad - r_{alkn}\text{NH4Nitr} + r_{alkden}\text{Denitr} \end{aligned} \tag{159}$$

where the *r*'s are ratios that translate the processes into the corresponding amount of alkalinity. The stoichiometric coefficients are derived from Equations (62) through (65) as in

Phytoplankton Photosynthesis (Ammonia as Substrate) and Respiration:

$$r_{alkaa} = r_{ca}\left(\frac{\text{gC}}{\text{mgA}}\right) \times \frac{14\,\text{eqH}^+}{106\,\text{moleC}} \times \frac{\text{moleC}}{12\,\text{gC}} \times \frac{\text{m}^3}{1000\,\text{L}} \tag{160}$$

Phytoplankton Photosynthesis (Nitrate as Substrate):

$$r_{alkan} = r_{ca}\left(\frac{\text{gC}}{\text{mgA}}\right) \times \frac{18\,\text{eqH}^+}{106\,\text{moleC}} \times \frac{\text{moleC}}{12\,\text{gC}} \times \frac{\text{m}^3}{1000\,\text{L}} \tag{161}$$

Bottom Algae Photosynthesis (Ammonia as Substrate) and Respiration:

$$r_{alkda} = r_{ca}\left(\frac{\text{gC}}{\text{gD}}\right) \times \frac{14\,\text{eqH}^+}{106\,\text{moleC}} \times \frac{\text{moleC}}{12\,\text{gC}} \times \frac{\text{m}^3}{1000\,\text{L}} \tag{162}$$

Bottom Algae Photosynthesis (Nitrate as Substrate):

$$r_{alkdn} = r_{ca}\left(\frac{\text{gC}}{\text{gD}}\right) \times \frac{18\,\text{eqH}^+}{106\,\text{moleC}} \times \frac{\text{moleC}}{12\,\text{gC}} \times \frac{\text{m}^3}{1000\,\text{L}} \tag{163}$$

Nitrification:

$$r_{alkn} = \frac{2\,\text{eqH}^+}{1\,\text{moleN}} \times \frac{\text{moleN}}{14\,\text{gN}} \times \frac{1\,\text{gN}}{1000\,\text{mgN}} \times \frac{1\,\text{m}^3}{1000\,\text{L}} \tag{164}$$

Denitrification:

$$r_{alkden} = \frac{4\,\text{eqH}^+}{1\,\text{moleN}} \times \frac{\text{moleN}}{14\,\text{gN}} \times \frac{1\,\text{gN}}{1000\,\text{mgN}} \times \frac{1\,\text{m}^3}{1000\,\text{L}} \tag{165}$$

5.4 SOD/Nutrient Flux Model

Sediment nutrient fluxes and sediment oxygen demand (SOD) are based on a model developed by Di Toro (Di Toro et al. 1991, Di Toro and Fitzpatrick. 1993, Di Toro 2001). The present version also benefited from James Martin's (Mississippi State University, personal communication) efforts to incorporate the Di Toro approach into EPA's WASP modeling framework.

A schematic of the model is depicted in Figure 21. As can be seen, the approach allows oxygen and nutrient sediment-water fluxes to be computed based on the downward flux of particulate organic matter from the overlying water. The sediments are divided into 2 layers: a thin (≅ 1 mm) surface aerobic layer underlain by a thicker (10 cm) lower anaerobic layer. Organic carbon, nitrogen and phosphorus are delivered to the anaerobic sediments via the settling of particulate organic matter (i.e., phytoplankton and detritus). There they are transformed by mineralization reactions into dissolved methane, ammonium and inorganic phosphorus. These constituents are then transported to the aerobic layer where some of the methane and ammonium are oxidized. The flux of oxygen from the water required for these oxidations is the sediment oxygen demand. The following sections provide details on how the model computes this SOD along with the sediment-water fluxes of carbon, nitrogen and phosphorus that are also generated in the process.

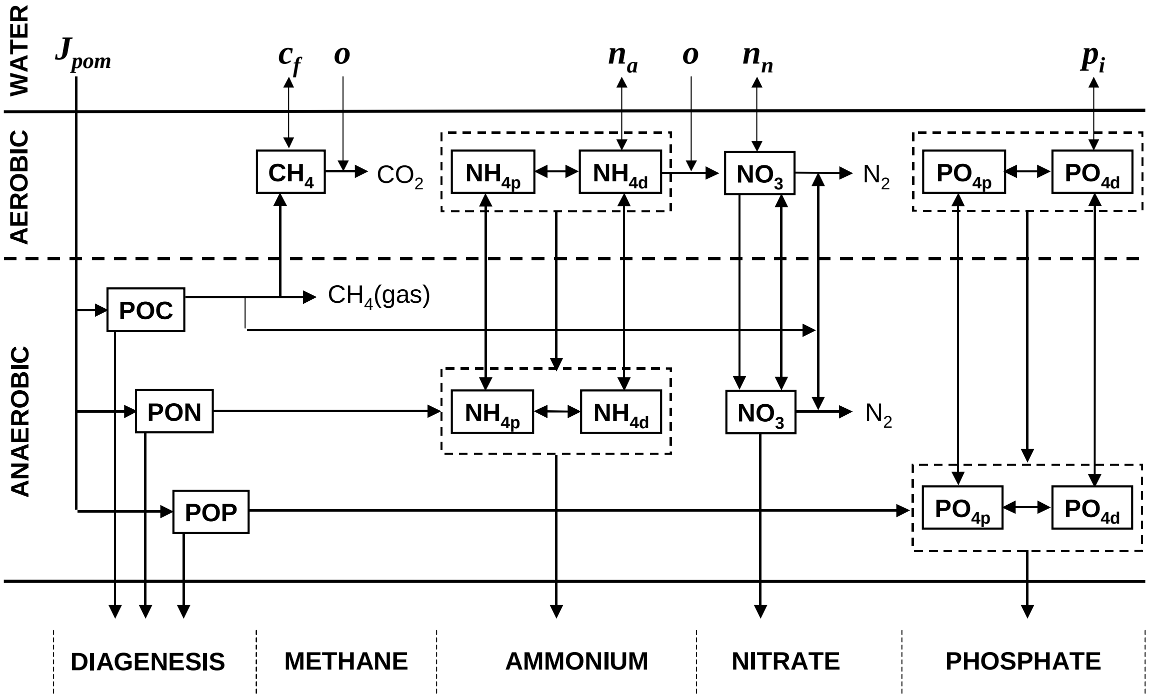


Figure 21 Schematic of SOD-nutrient flux model of the sediments.

5.4.1 Diagenesis

As summarized in Figure 22, the first step in the computation involves determining how much of the downward flux of particulate organic matter (POM) is converted into soluble reactive forms in the anaerobic sediments. This process is referred to as diagenesis. First the total downward flux is computed as the sum of the fluxes of phytoplankton and detritus settling from the water column

$$J_{POM} = r_{da} v_a a_p + v_{dt} m_o \tag{166}$$

where J_{POM} = the downward flux of POM [gD m^{-2} d^{-1}], r_{da} = the ratio of dry weight to chlorophyll a [gD/mgA], v_a = phytoplankton settling velocity [m/d], a_p = phytoplankton concentration [mgA/m^3], v_{dt} = detritus settling velocity [m/d], and m_o = detritus concentration [gD/m^3].

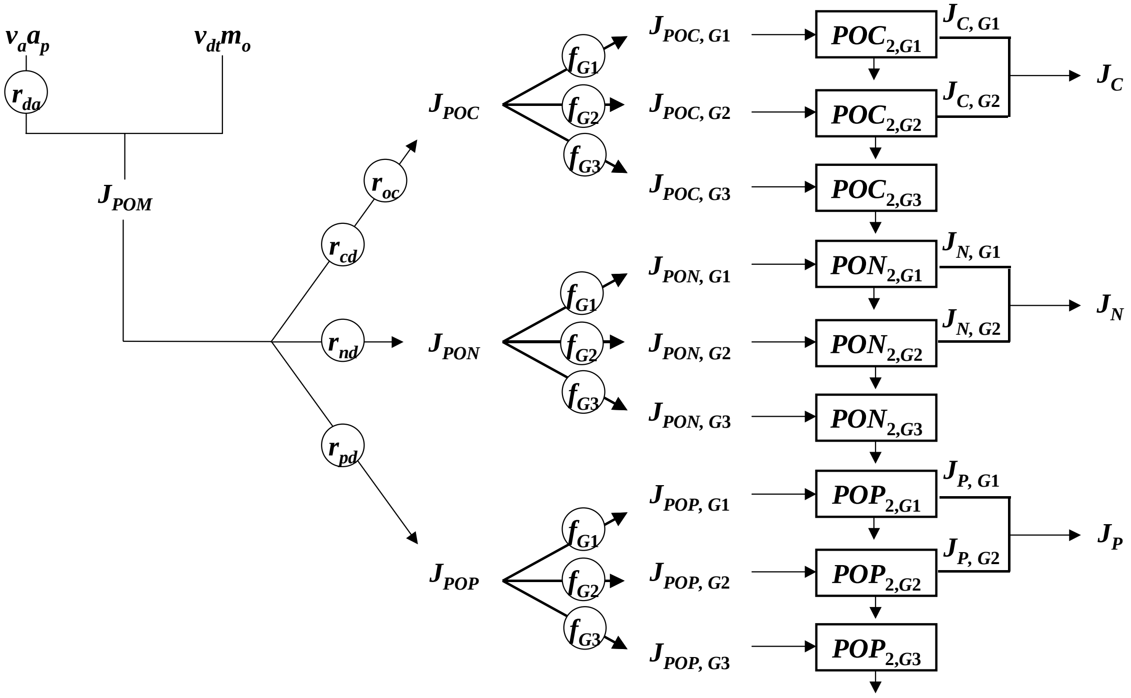


Figure 22 Representation of how settling particulate organic particles (phytoplankton and detritus) are transformed into fluxes of dissolved carbon (J_C) , nitrogen (J_N) and phosphorus (J_P) in the anaerobic sediments.

Stoichiometric ratios are then used to divide the POM flux into carbon, nitrogen and phosphorus. Note that for convenience, we will express the particulate organic carbon (POC) as oxygen equivalents using the stoichiometric coefficient r_{oc}. Each of the nutrient fluxes is further broken down into three reactive fractions: labile (G1), slowly reacting (G2) and non-reacting (G3).

These fluxes are then entered into mass balances to compute the concentration of each fraction in the anaerobic layer. For example, for labile POC, a mass balance is written as

$$H_2 \frac{dPOC_{2,G1}}{dt} = J_{POC,G1} - k_{POC,G1}\theta_{POC,G1}^{T-20} H_2 POC_{2,G1} - w_2 POC_{2,G1} \tag{167}$$

where H_2 = the thickness of the anaerobic layer [m], $POC_{2,G1}$ = the concentration of the labile fraction of POC in the anaerobic layer [gO$_2$/m^3], $J_{POC,G1}$ = the flux of labile POC delivered to the anaerobic layer [gO$_2$/m^2/d], $k_{POC,G1}$ = the mineralization rate of labile POC [d^{-1}], $\theta_{POC,G1}$ = temperature correction factor for labile POC mineralization [dimensionless], and w_2 = the burial velocity [m/d]. At steady state, Eq. 166 can be solved for

$$POC_{2,G1} = \frac{J_{POC,G1}}{k_{POC,G1}\theta_{POC,G1}^{T-20} H_2 + v_b} \tag{168}$$

The flux of labile dissolved carbon, $J_{C,G1}$ [gO$_2$/m^2/d], can then be computed as

$$J_{C,G1} = k_{POC,G1}\theta_{POC,G1}^{T-20} H_2 POC_{2,G1} \tag{169}$$

In a similar fashion, a mass balance can be written and solved for the slowly reacting dissolved organic carbon. This result is then added to Eq. 168 to arrive at the total flux of dissolved carbon generated in the anaerobic sediments.

$$J_C = J_{C,G1} + J_{C,G2} \tag{170}$$

Similar equations are developed to compute the diagenesis fluxes of nitrogen, J_N [gN/m^2/d], and phosphorus J_P [gP/m^2/d].

5.4.2 Ammonium

Based on the mechanisms depicted in Figure 21, mass balances can be written for total ammonium in the aerobic layer and the anaerobic layers,

$$\begin{aligned} H_1 \frac{dNH_{4,1}}{dt} &= \omega_{12}\left(f_{pa2} NH_{4,2} - f_{pa1} NH_{4,1}\right) + K_{L12}\left(f_{da2} NH_{4,2} - f_{da1} NH_{4,1}\right) - w_2 NH_{4,1} \\ &\quad + s\left(\frac{n_a}{1000} - f_{da1} NH_{4,1}\right) - \frac{\kappa_{NH4,1}^2}{s}\theta_{NH4}^{T-20} \frac{K_{NH4}}{K_{NH4} + NH_{4,1}} \frac{o}{2K_{NH4,O2} + o} f_{da1} NH_{4,1} \end{aligned} \tag{171}$$

$$\begin{aligned} H_2 \frac{dNH_{4,2}}{dt} &= J_N + \omega_{12}\left(f_{pa1} NH_{4,1} - f_{pa2} NH_{4,2}\right) + K_{L12}\left(f_{da1} NH_{4,1} - f_{da2} NH_{4,2}\right) \\ &\quad + w_2\left(NH_{4,1} - NH_{4,2}\right) \end{aligned} \tag{172}$$

where H_1 = the thickness of the aerobic layer [m], $NH_{4,1}$ and $NH_{4,2}$ = the concentration of total ammonium in the aerobic layer and the anaerobic layers, respectively [gN/m^3], n_a = the ammonium concentration in the overlying water [mgN/m^3], $\kappa_{NH4,1}$ = the reaction velocity for nitrification in the aerobic sediments [m/d], θ_{NH4} = temperature correction factor for nitrification [dimensionless], K_{NH4} = ammonium half-saturation constant [gN/m^3], o = the dissolved oxygen concentration in the overlying water [gO_2/m^3], and $K_{NH4,O2}$ = oxygen half-saturation constant [mgO_2/L], and J_N = the diagenesis flux of ammonium [$gN/m^2/d$].

The fraction of ammonium in dissolved (f_{dai}) and particulate (f_{pai}) form are computed as

$$f_{dai} = \frac{1}{1 + m_i \pi_{ai}} \tag{173}$$

$$f_{pai} = 1 - f_{dai} \tag{174}$$

where m_i = the solids concentration in layer i [gD/m^3], and π_{ai} = the partition coefficient for ammonium in layer i [m^3/gD].

The mass transfer coefficient for particle mixing due to bioturbation between the layers, ω_{12} [m/d], is computed as

$$\omega_{12} = \frac{D_p \theta_{Dp}^{T-20}}{H_2} \frac{POC_{2,G1} / r_{oc}}{POC_R} \frac{o}{K_{M,Dp} + o} \tag{175}$$

where D_p = diffusion coefficient for bioturbation [m^2/d], θ_{Dp} = temperature coefficient [dimensionless], POC_R = reference G1 concentration for bioturbation [gC/m^3] and $K_{M,Dp}$ = oxygen half-saturation constant for bioturbation [gO_2/m^3].

The mass transfer coefficient for pore water diffusion between the layers, K_{L12} [m/d], is computed as,

$$K_{L12} = \frac{D_d \theta_{Dd}^{T-20}}{H_2 / 2} \tag{176}$$

where D_d = pore water diffusion coefficient [m^2/d], and θ_{Dd} = temperature coefficient [dimensionless].

The mass transfer coefficient between the water and the aerobic sediments, s [m/d], is computed as

$$s = \frac{SOD}{o} \tag{177}$$

where SOD = the sediment oxygen demand [$gO_2/m^2/d$].

At steady state, Eqs. 170 and 171 are two simultaneous nonlinear algebraic equations. The equations can be linearized by assuming that the $NH_{4,1}$ term in the Monod term for nitrification is constant. The simultaneous linear equations can then be solved for $NH_{4,1}$ and $NH_{4,2}$. The flux of ammonium to the overlying water can then be computed as

$$J_{NH4} = s\left(f_{da1} NH_{4,1} - \frac{n_a}{1000} \right) \tag{178}$$

5.4.3 Nitrate

Mass balances for nitrate can be written for the aerobic and anaerobic layers as

$$H_1 \frac{dNO_{3,1}}{dt} = K_{L12}\left(NO_{3,2} - NO_{3,1}\right) - w_2 NO_{3,1} + s\left(\frac{n_n}{1000} - NO_{3,1}\right) \\ + \frac{\kappa_{NH4,1}^2}{s}\theta_{NH4}^{T-20}\frac{K_{NH4}}{K_{NH4} + NH_{4,1}}\frac{o}{2K_{NH4,O2} + o} f_{da1} NH_{4,1} - \frac{\kappa_{NO3,1}^2}{s}\theta_{NO3}^{T-20} NO_{3,1} \tag{179}$$

$$H_2 \frac{dNO_{3,2}}{dt} = J_N + K_{L12}\left(NO_{3,1} - NO_{3,2}\right) + w_2\left(NO_{3,1} - NO_{3,2}\right) - \kappa_{NO3,2}\theta_{NO3}^{T-20} NO_{3,2} \tag{180}$$

where $NO_{3,1}$ and $NO_{3,2}$ = the concentration of nitrate in the aerobic layer and the anaerobic layers, respectively [gN/m^3], n_n = the nitrate concentration in the overlying water [mgN/m^3], $\kappa_{NO3,1}$ and $\kappa_{NO3,2}$ = the reaction velocities for denitrification in the aerobic and anaerobic sediments, respectively [m/d], and θ_{NO3} = temperature correction factor for denitrification [dimensionless].

In the same fashion as for Eqs. 170 and 171, Eqs. 178 and 179 can be linearized and solved for $NO_{3,1}$ and $NO_{3,2}$. The flux of nitrate to the overlying water can then be computed as

$$J_{NO3} = s\left(NO_{3,1} - \frac{n_n}{1000}\right) \tag{181}$$

Denitrification requires a carbon source as represented by the following chemical equation,

$$5\mathrm{CH_2O} + 4\mathrm{NO_3^-} + 4\mathrm{H^+} \rightarrow 5\mathrm{CO_2} + 2\mathrm{N_2} + 7\mathrm{H_2O} \tag{182}$$

The carbon requirement (expressed in oxygen equivalents per nitrogen) can therefore be computed as

$$r_{ondn} = 2.67\frac{\mathrm{gO_2}}{\mathrm{gC}}\frac{5\ \mathrm{moleC} \times 12\ \mathrm{gC/moleC}}{4\ \mathrm{moleN} \times 14\ \mathrm{gN/moleN}} \times \frac{1\ \mathrm{gN}}{1000\ \mathrm{mgN}} = 0.00286\frac{\mathrm{gO_2}}{\mathrm{mgN}} \tag{183}$$

Therefore, the oxygen equivalents consumed during denitrification, $J_{O2,dn}$ [gO$_2$/m^2/d], can be computed as

$$J_{O2,dn} = 1000\frac{\mathrm{mgN}}{\mathrm{gN}} \times r_{ondn}\left(\frac{\kappa_{NO3,1}^2}{s}\theta_{NO3}^{T-20} NO_{3,1} + \kappa_{NO3,2}\theta_{NO3}^{T-20} NO_{3,2}\right) \tag{184}$$

5.4.4 Methane

The dissolved carbon generated by diagenesis is converted to methane in the anaerobic sediments. Because methane is relatively insoluble, its saturation can be exceeded and methane gas produced. As a consequence, rather than write a mass balance for methane in the anaerobic layer, an analytical model developed by Di Toro et al. (1990) is used to determine the steady-state flux of dissolved methane corrected for gas loss delivered to the aerobic sediments.

First, the carbon diagenesis flux is corrected for the oxygen equivalents consumed during denitrification,

$$J_{CH4,T} = J_C - J_{O2,dn} \tag{185}$$

where $J_{CH4,T}$ = the carbon diagenesis flux corrected for denitrification [gO$_2$/m^2/d]. In other words, this is the total anaerobic methane production flux expressed in oxygen equivalents.

If $J_{CH4,T}$ is sufficiently large ($\geq 2K_{L12}C_s$), methane gas will form. In such cases, the flux can be corrected for the gas loss,

$$J_{CH4,d} = \sqrt{2K_{L12}C_s J_{CH4,T}} \tag{186}$$

where $J_{CH4,d}$ = the flux of dissolved methane (expressed in oxygen equivalents) that is generated in the anaerobic sediments and delivered to the aerobic sediments [gO$_2$/m^2/d], C_s = the saturation concentration of methane expressed in oxygen equivalents [mgO$_2$/L]. If $J_{CH4,T} < 2K_{L12}C_s$, then no gas forms and

$$J_{CH4,d} = J_{CH4,T} \tag{187}$$

The methane saturation concentration is computed as

$$C_s = 100\left(1 + \frac{H}{10}\right)1.024^{20-T} \tag{188}$$

where H = water depth [m] and T = water temperature [$^{\circ}$C].

A methane mass balance can then be written for the aerobic layer as

$$H_1 \frac{dCH_{4,1}}{dt} = J_{CH4,d} + s\left(c_f - CH_{4,1}\right) - \frac{\kappa_{CH4,1}^2}{s}\theta_{CH4}^{T-20} CH_{4,1} \tag{189}$$

where $CH_{4,1}$ = methane concentration in the aerobic layer [gO$_2$/m^3], c_f = fast CBOD in the overlying water [gO$_2$/m^3], $\kappa_{CH4,1}$ = the reaction velocity for methane oxidation in the aerobic sediments [m/d], and θ_{CH4} = temperature correction factor [dimensionless]. At steady, state, this balance can be solved for

$$CH_{4,1} = \frac{J_{CH4,d} + sc_f}{s + \dfrac{\kappa_{CH4,1}^2}{s}\theta_{CH4}^{T-20}} \tag{190}$$

The flux of methane to the overlying water, J_{CH4} [gO$_2$/m^2/d], can then be computed as

$$J_{CH4} = s\left(CH_{4,1} - c_f\right) \tag{191}$$

5.4.5 SOD

The *SOD* [$gO_2/m^2/d$] is equal to the sum of the oxygen consumed in methane oxidation and nitrification,

$$SOD = CSOD + NSOD \tag{192}$$

where *CSOD* = the amount of oxygen demand generated by methane oxidation [$gO_2/m^2/d$] and *NSOD* = the amount of oxygen demand generated by nitrification [$gO_2/m^2/d$]. These are computed as

$$CSOD = \frac{\kappa_{CH4,1}^2}{s} \theta_{CH4}^{T-20} CH_{4,1} \tag{193}$$

$$NSOD = r_{on} \frac{\kappa_{NH4,1}^2}{s} \theta_{NH4}^{T-20} \frac{K_{NH4}}{K_{NH4} + NH_{4,1}} \frac{o}{2K_{NH4,O2} + o} f_{da1} NH_{4,1} \tag{194}$$

where r_{on} = the ratio of oxygen to nitrogen consumed during nitrification [= 4.57 gO_2/gN].

5.4.6 Inorganic Phosphorus

Mass balances can be written total inorganic phosphorus in the aerobic layer and the anaerobic layers as

$$\begin{aligned} H_1 \frac{dPO_{4,1}}{dt} = \omega_{12}\left(f_{pp2} PO_{4,2} - f_{pp1} PO_{4,1}\right) + K_{L12}\left(f_{dp2} PO_{4,2} - f_{dp1} PO_{4,1}\right) \\ - w_2 PO_{4,1} + s\left(\frac{p_i}{1000} - f_{da1} PO_{4,1}\right) \end{aligned} \tag{195}$$

$$\begin{aligned} H_2 \frac{dPO_{4,2}}{dt} = J_P + \omega_{12}\left(f_{pp1} PO_{4,1} - f_{pp2} PO_{4,2}\right) + K_{L12}\left(f_{dp1} PO_{4,1} - f_{dp2} PO_{4,2}\right) \\ + w_2\left(PO_{4,1} - PO_{4,2}\right) \end{aligned} \tag{196}$$

where $PO_{4,1}$ and $PO_{4,2}$ = the concentration of total inorganic phosphorus in the aerobic layer and the anaerobic layers, respectively [gP/m^3], p_i = the inorganic phosphorus in the overlying water [mgP/m^3], and J_P = the diagenesis flux of phosphorus [$gP/m^2/d$].

The fraction of phosphorus in dissolved (f_{dpi}) and particulate (f_{ppi}) form are computed as

$$f_{dpi} = \frac{1}{1 + m_i \pi_{pi}} \tag{197}$$

$$f_{ppi} = 1 - f_{dpi} \tag{198}$$

where π_{pi} = the partition coefficient for inorganic phosphorus in layer i [m^3/gD].

The partition coefficient in the anaerobic layer is set to an input value. For the aerobic layer, if the oxygen concentration in the overlying water column exceeds a critical concentration, o_{crit} [gO$_2$/m^3], then the partition coefficient is increased to represent the sorption of phosphorus onto iron oxyhydroxides as in

$$\pi_{p1} = \pi_{p2}\left(\Delta\pi_{PO4,1}\right) \tag{199}$$

where $\Delta\pi_{PO4,1}$ is a factor that increases the aerobic layer partition coefficient relative to the anaerobic coefficient.

If the oxygen concentration falls below o_{crit} then the partition coefficient is decreased smoothly until it reaches the anaerobic value at zero oxygen,

$$\pi_{p1} = \pi_{p2}\left(\Delta\pi_{PO4,1}\right)^{o/o_{crit}} \tag{200}$$

Equations 194 and 195 can be solved for $PO_{4,1}$ and $PO_{4,2}$. The flux of phosphorus to the overlying water can then be computed as

$$J_{PO4} = s\left(PO_{4,1} - \frac{p_i}{1000}\right) \tag{201}$$

5.4.7 Solution Scheme

Although the foregoing sequence of equations can be solved, a single computation will not yield a correct result because of the interdependence of the equations. For example, the surface mass transfer coefficient s depends on SOD. The SOD in turn depends on the ammonium and methane concentrations which themselves are computed via mass balances that depend on s. Hence, an iterative technique must be used. The procedure used in QUAL2K is

1. Determine the diagenesis fluxes: J_C, J_N and J_P.

2. Start with an initial estimate of SOD,

   $$SOD_{init} = J_C + r_{on}^{'} J_N \tag{202}$$

   where r_{on}' = the ratio of oxygen to nitrogen consumed for total conversion of ammonium to nitrogen gas via nitrification/denitrification [= 1.714 gO$_2$/gN]. This ratio accounts for the carbon utilized for denitrification.

3. Compute s using

   $$s = \frac{SOD_{init}}{o} \tag{203}$$

4. Solve for ammonium, nitrate and methane, and compute the $CSOD$ and $NSOD$.

5. Make a revised estimate of SOD using the following weighted average

$$SOD = \frac{SOD_{init} + CSOD + NSOD}{2} \tag{204}$$

6. Check convergence by calculating an approximate relative error

$$\varepsilon_a = \left| \frac{SOD - SOD_{init}}{SOD} \right| \times 100\% \tag{205}$$

7. If ε_a is greater than a prespecified stopping criterion ε_s then set $SOD_{init} = SOD$ and return to step 2.

8. If convergence is adequate ($\varepsilon_a \le \varepsilon_s$), then compute the inorganic phosphorus concentrations.

9. Compute the ammonium, nitrate, methane and phosphate fluxes.

5.4.8 Supplementary Fluxes

Because of the presence of organic matter deposited prior to the summer steady-state period (e.g., during spring runoff), it is possible that the downward flux of particulate organic matter is insufficient to generate the observed SOD. In such cases, a supplementary SOD can be prescribed,

$$SOD_t = SOD + SOD_s \tag{206}$$

where SOD_t = the total sediment oxygen demand [$gO_2/m^2/d$], and SOD_s = the supplementary SOD [$gO_2/m^2/d$]. In addition, prescribed ammonia and methane fluxes can be used to supplement the computed fluxes.

6 USERS MANUAL

6.1 OVERVIEW

The computer code used to implement the calculations for QUAL2K is written in Visual Basic for Applications (VBA). Excel serves as the user interface.

Color is used to signify whether information is to be input by the user or output by the program:

- **Pale Blue** designates variable and parameter values that are to be entered by the user.
- **Pale Yellow** designates data that the user enters. This data are then displayed on graphs generated by Q2K.
- **Pale Green** designates output values generated by Q2K.
- **Dark solid colors** are used for labels and should not be changed.

All worksheets include two buttons (Figure 23):

- **Open Old Files.** When this button is clicked, the file browser will automatically open to allow you to access a data file. All QUAL2K data files have the extension, *.q2k.
- **Run.** This button causes Q2K to execute and to create a data file that holds the input values. The data file can then be accessed later using the **Open Old File** button.

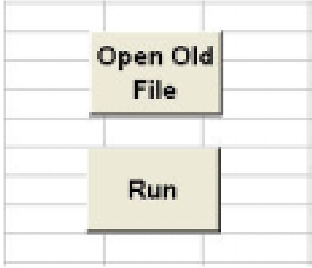
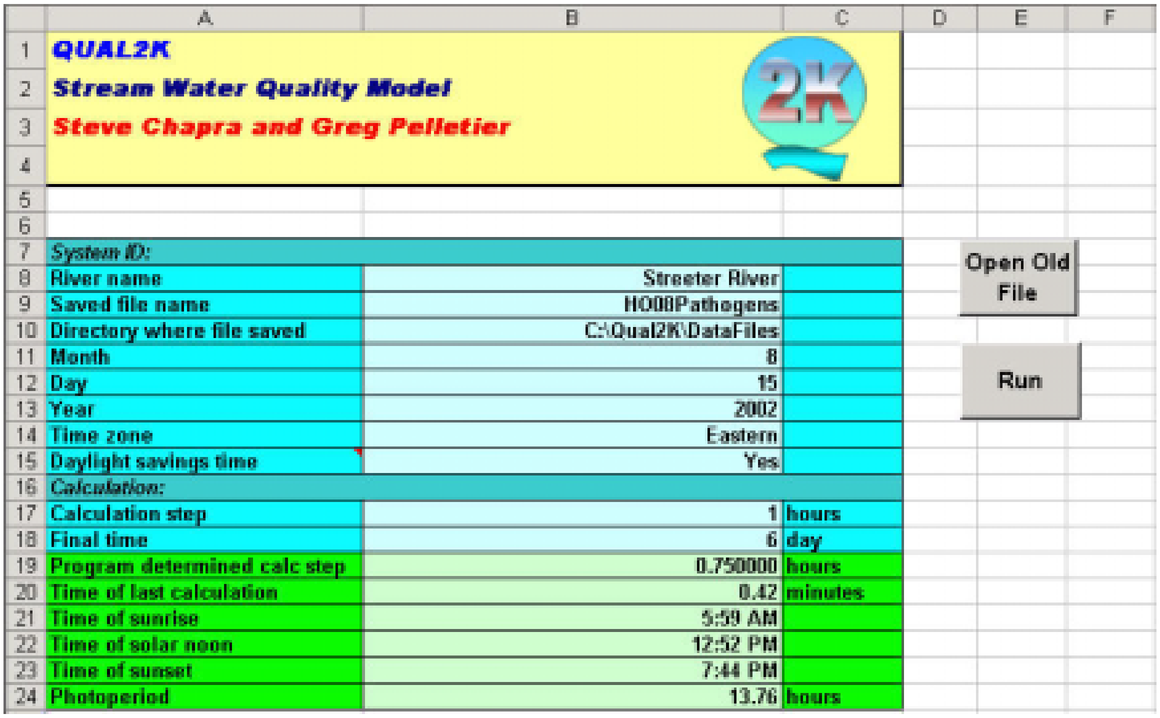


Figure 23. The buttons used in Q2K.

6.2 WORKSHEETS

6.2.1 QUAL2K Worksheet

The QUAL2K Worksheet (Figure 24) is used to enter general information regarding a particular model application.



System ID:		
River name	Streeter River	
Saved file name	HO00Pathogens	
Directory where file saved	C:\Qual2K\DataFiles	
Month	8	
Day	15	
Year	2002	
Time zone	Eastern	
Daylight savings time	Yes	
Calculation:		
Calculation step	1	hours
Final time	6	day
Program determined calc step	0.750000	hours
Time of last calculation	0.42	minutes
Time of sunrise	5:59 AM	
Time of solar noon	12:52 PM	
Time of sunset	7:44 PM	
Photoperiod	13.76	hours

Figure 24. The QUAL2K Worksheet.

River name. Name of the river or stream being modeled. After the program is run, this name along with the date, is displayed on all sheets and charts.
File name. This is the name of the data file generated when Q2K is run.
Directory where file saved. This specifies the complete path to the directory where the file is saved.
Month. The simulation month. This is entered in numerical format (e.g., January = 1, February = 2, etc.).
Day. The simulation day.
Year. The simulation year (e.g., 1993)
Time zone. A pull-down menu (Figure 25) allows you to select the proper U.S. time zone.

Figure 25. The pull down menu for setting the time zone.

Daylight savings time. A pull down menu allows you to specify whether daylight savings time is in effect (Yes or No).

Calculation step. This is the time step used for the calculation. It must be less than or equal to 4 hrs. If the user enters a value greater than 4 hours, the program automatically sets the calculation step to 4 hours. This level of resolution is required in order to have a sufficient number of points to create smooth diel plots.

Final time. This defines the duration of the calculation. It must be an integer that is greater than or equal to 2 days. This constraint is imposed because the model is run in a time variable mode until it reaches a steady state. Therefore, the first day of simulation is by definition overwhelmingly dominated by its initial conditions. If the user enters a value less than 2 days, the program automatically sets the final time to 2 days. The final time should be at least twice the river's travel time. For streams with short travel times where bottom algae are simulated, it must usually be longer.

Program determined calc step (output). The program takes the **Calculation step** entered by the user and then rounds it down to the next lowest whole base-2 number. In order to use a lower time step, you must reduce the calculation step below this value.

Time of last calculation (output). The computer automatically displays the computer time required for the simulation.

Time of sunrise. This is the time of sunrise for the farthest downstream reach.

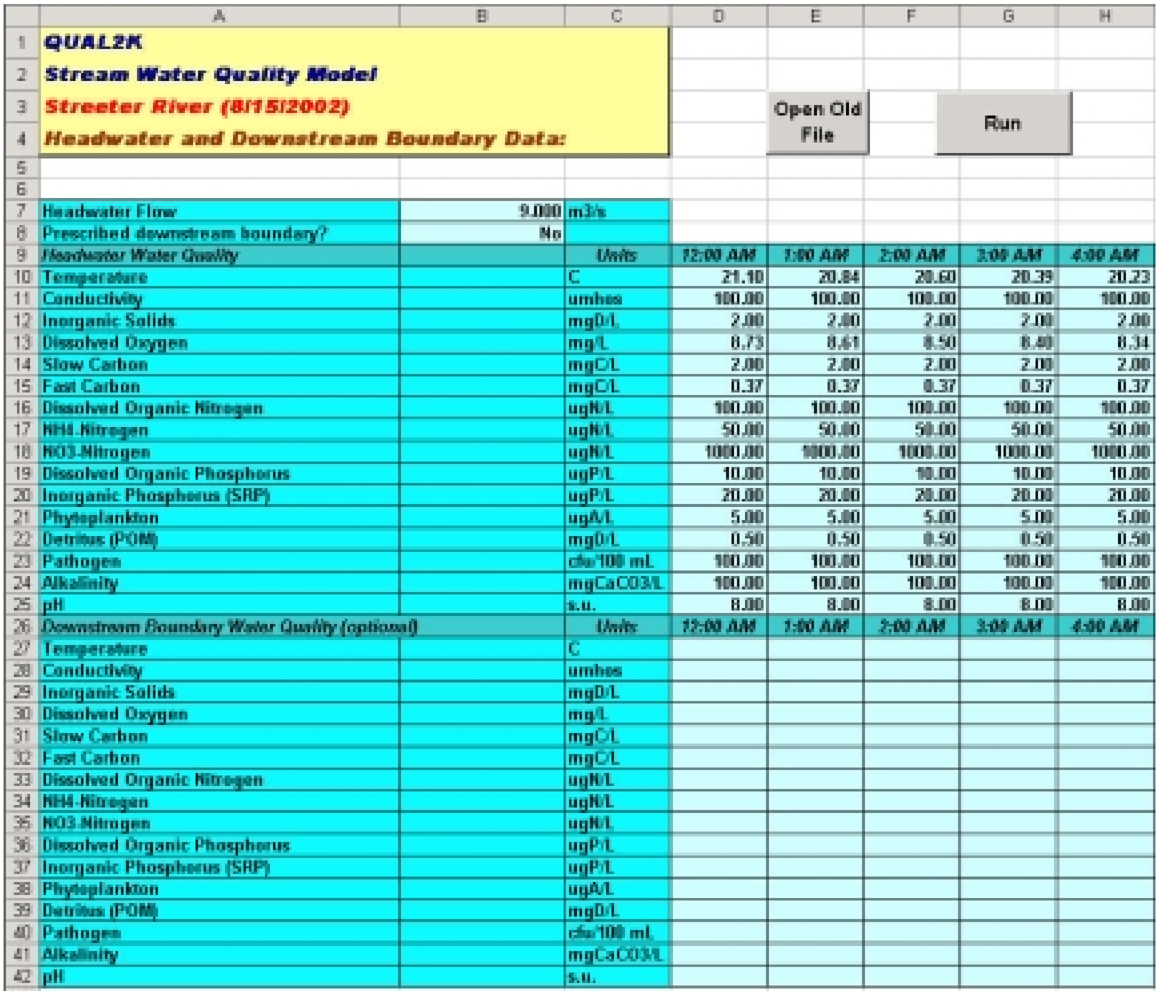
Time of solar noon. This is the time of solar noon for the farthest downstream reach.

Time of sunset. This is the time of sunset for the farthest downstream reach.

Photoperiod. This is the fraction of the day that the sun is up for the farthest downstream reach. It is equal to the time in hours between sunrise and sunset divided by 24.

6.2.2 Headwater Worksheet

This worksheet (Figure 26) is used to enter flow and concentration for the system's boundaries.



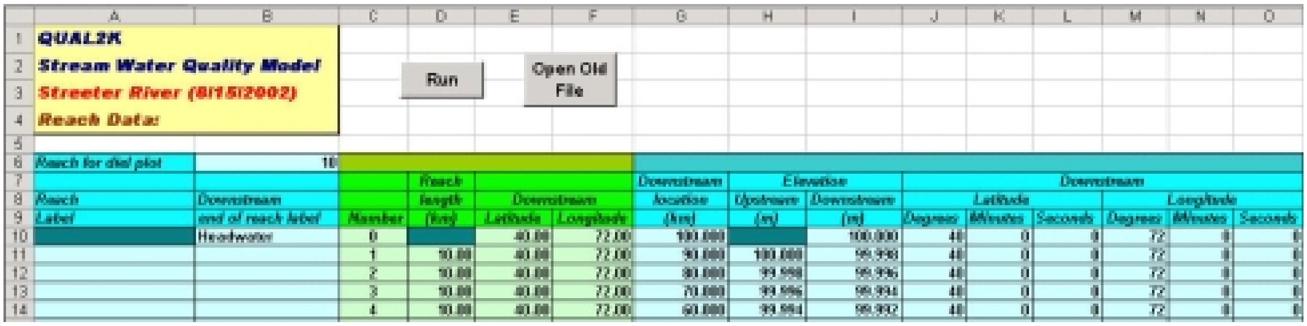
	A	B	C	D	E	F	G	H
1	***QUAL2K***							
2	***Stream Water Quality Model***							
3	***Streeter River (8/15/2002)***							
4	***Headwater and Downstream Boundary Data:***							
5								
6								
7	Headwater Flow	9.000	m3/s					
8	Prescribed downstream boundary?	No						
9	*Headwater Water Quality*		*Units*	*12:00 AM*	*1:00 AM*	*2:00 AM*	*3:00 AM*	*4:00 AM*
10	Temperature		C	21.10	20.84	20.60	20.39	20.23
11	Conductivity		umhos	100.00	100.00	100.00	100.00	100.00
12	Inorganic Solids		mgD/L	2.00	2.00	2.00	2.00	2.00
13	Dissolved Oxygen		mg/L	8.73	8.61	8.50	8.40	8.34
14	Slow Carbon		mgC/L	2.00	2.00	2.00	2.00	2.00
15	Fast Carbon		mgC/L	0.37	0.37	0.37	0.37	0.37
16	Dissolved Organic Nitrogen		ugN/L	100.00	100.00	100.00	100.00	100.00
17	NH4-Nitrogen		ugN/L	50.00	50.00	50.00	50.00	50.00
18	NO3-Nitrogen		ugN/L	1000.00	1000.00	1000.00	1000.00	1000.00
19	Dissolved Organic Phosphorus		ugP/L	10.00	10.00	10.00	10.00	10.00
20	Inorganic Phosphorus (SRP)		ugP/L	20.00	20.00	20.00	20.00	20.00
21	Phytoplankton		ugA/L	5.00	5.00	5.00	5.00	5.00
22	Detritus (POM)		mgD/L	0.50	0.50	0.50	0.50	0.50
23	Pathogen		cfu/100 mL	100.00	100.00	100.00	100.00	100.00
24	Alkalinity		mgCaCO3/L	100.00	100.00	100.00	100.00	100.00
25	pH		s.u.	8.00	8.00	8.00	8.00	8.00
26	*Downstream Boundary Water Quality (optional)*		*Units*	*12:00 AM*	*1:00 AM*	*2:00 AM*	*3:00 AM*	*4:00 AM*
27	Temperature		C					
28	Conductivity		umhos					
29	Inorganic Solids		mgD/L					
30	Dissolved Oxygen		mg/L					
31	Slow Carbon		mgC/L					
32	Fast Carbon		mgC/L					
33	Dissolved Organic Nitrogen		ugN/L					
34	NH4-Nitrogen		ugN/L					
35	NO3-Nitrogen		ugN/L					
36	Dissolved Organic Phosphorus		ugP/L					
37	Inorganic Phosphorus (SRP)		ugP/L					
38	Phytoplankton		ugA/L					
39	Detritus (POM)		mgD/L					
40	Pathogen		cfu/100 mL					
41	Alkalinity		mgCaCO3/L					
42	pH		s.u.					

Figure 26. The upper part of the Headwater Worksheet used to enter the headwater's constant characteristics and to specify its hydraulics.

Flow. The headwater's flow rate in m^3/s.
Prescribed Downstream Boundary? If the downstream boundary has an effect on the simulation, this option is set to yes.
Headwater Water Quality. This block of cells is used to enter the temperature and water quality boundary conditions at the river's headwater. For cases where the data varies in a diel fashion, Q2K allows you to enter values on an hourly basis. If the values are constant over the daily cycle, just enter the mean value for all times.
Downstream Boundary Water Quality. If the downstream boundary has an effect on the simulation, this block of cells is used to enter the temperature and water quality conditions at the river's downstream boundary. headwater.

6.2.3 Reach Worksheet

This worksheet is used to enter information related to the river's headwater (Reach Number 0) and reaches (Figure 27 through Figure 31).



QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Reach Data:

Reach for diel plot	10													
			Reach			Downstream	Elevation		Downstream					
Reach	Downstream		length	Downstream		location	Upstream	Downstream	Latitude			Longitude		
Label	end of reach label	Number	(km)	Latitude	Longitude	(km)	(m)	(m)	Degrees	Minutes	Seconds	Degrees	Minutes	Seconds
	Headwater	0		40.00	72.00	100.000		100.000	40	0	0	72	0	0
		1	10.00	40.00	72.00	90.000	100.000	99.998	40	0	0	72	0	0
		2	10.00	40.00	72.00	80.000	99.998	99.996	40	0	0	72	0	0
		3	10.00	40.00	72.00	70.000	99.996	99.994	40	0	0	72	0	0
		4	10.00	40.00	72.00	60.000	99.994	99.992	40	0	0	72	0	0

Figure 27. The first part of the Reach Worksheet used to specify reach labels, distances and elevations.

Reach for diel plot. Cell B6 is used to enter the number of the reach for which diel plots will be generated. If a negative, zero or a value greater than the number of reaches is entered, the program automatically sets the value to the last downstream reach.
Reach Label (optional). Q2K allows you to enter identification labels for each reach. Figure 28 provides an example to illustrate the naming scheme. The first two reaches of a river are shown. Because it includes the Jefferson City WWTP discharge, we might choose to enter the reach label "Jefferson City WWTP" for the first reach. Similarly we might label the second reach as "Sampling Station 27."

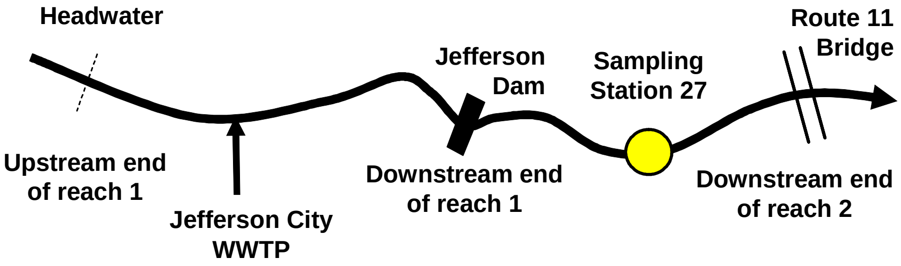
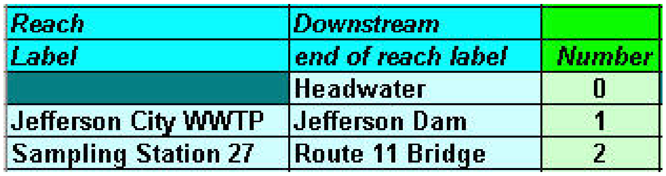


Figure 28. The first two reaches of a river system.

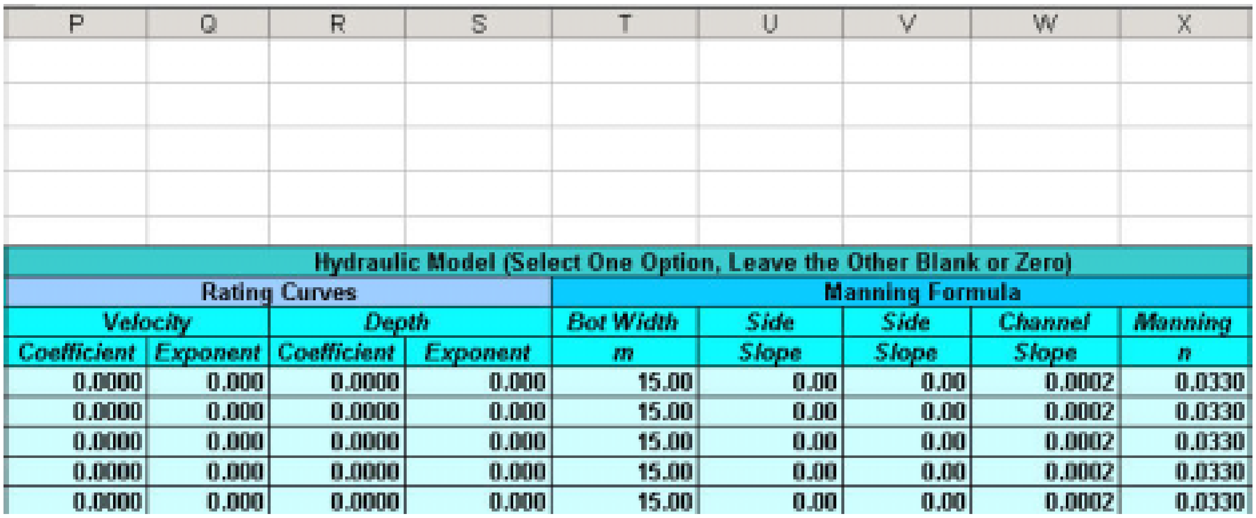
Downstream end of reach label (optional). Q2K allows you to enter identification labels for the boundaries between reaches. These labels are then displayed on other worksheets to identify the reaches. As shown in Figure 29, the downstream end of the first reach in Figure 28 could be labeled as "Jefferson Dam". Similarly, the downstream end of the second reach could be labeled as "Route 11 Bridge".



Reach	Downstream	
Label	end of reach label	Number
	Headwater	0
Jefferson City WWTP	Jefferson Dam	1
Sampling Station 27	Route 11 Bridge	2

Figure 29. An example of the labels that could be entered for the reaches in Figure 28.

Reach numbers (output). The model automatically numbers the reaches in ascending order.
Reach length (output). The model automatically computes and displays the length of each reach.
Downstream Latitude and Longitude (output). The model automatically computes and displays the latitude and longitude of the downstream ends of each reach in decimal degrees.
Downstream location. The user must enter the river kilometer for the downstream end of each reach. Note that the reach distances can be in descending or ascending order.
Upstream and downstream elevation. The user must enter the elevation in meters above sea level for both the upstream and downstream ends of the reach.
Downstream Latitude and Longitude (output). The user must enter the latitude and longitude of the downstream end of each reach in degrees, minutes, and seconds. Alternatively, they can be entered in decimal degrees, in which case, the minutes and seconds entries would be left blank or zero.



P	Q	R	S	T	U	V	W	X
Hydraulic Model (Select One Option, Leave the Other Blank or Zero)								
Rating Curves				Manning Formula				
Velocity		Depth		Bot Width	Side	Side	Channel	Manning
Coefficient	Exponent	Coefficient	Exponent	m	Slope	Slope	Slope	n
0.0000	0.000	0.0000	0.000	15.00	0.00	0.00	0.0002	0.0330
0.0000	0.000	0.0000	0.000	15.00	0.00	0.00	0.0002	0.0330
0.0000	0.000	0.0000	0.000	15.00	0.00	0.00	0.0002	0.0330
0.0000	0.000	0.0000	0.000	15.00	0.00	0.00	0.0002	0.0330
0.0000	0.000	0.0000	0.000	15.00	0.00	0.00	0.0002	0.0330

Figure 30. The part of the Reach Worksheet used to specify the system's hydraulics.

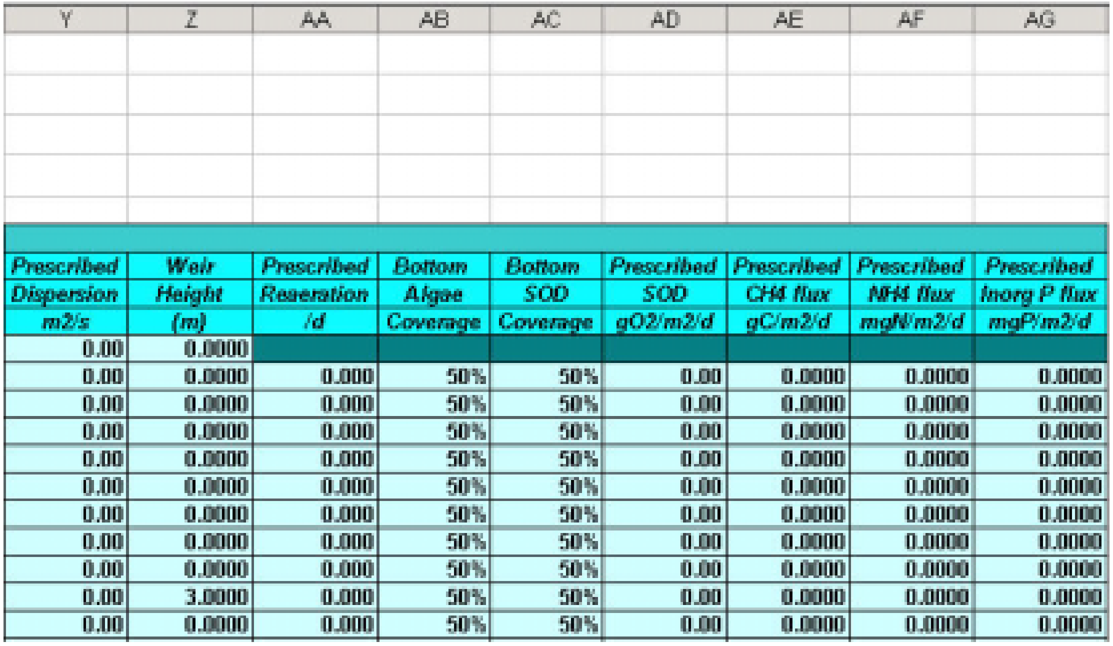
Hydraulic Model. Q2K allows two options for computing velocity and depth based on flow: (1) rating curves or (2) the Manning formula. **It is important to pick one of the options and leave the other blank or zero.** If the model detects a blank or zero value for the Manning *n*, it will implement the rating curves. Otherwise, the Manning formula will be solved.
Rating Curves: Described in Sec. 2.2.1.

- **Velocity coefficient.** a
- **Velocity exponent.** b
- **Depth coefficient.** α
- **Depth exponent.** β

Manning Formula: Described in Sec. 2.2.2

- **Bottom width.** The reach's bottom width, B_0 (m).
- **Side slope.** Number must be greater than zero. For example, a rectangular channel would have both side slopes equal to zero.
- **Channel slope.** The slope of the channel in meter of drop per meter of distance.
- **Manning n.** Dimensionless number that parameterizes channel roughness. Values for weedless man-made canals range from 0.012 to 0.03 and for natural channels from 0.025 to

0.2 (see Table 2 for additional details). A value of 0.04 is a good starting value for many natural channels.



Y	Z	AA	AB	AC	AD	AE	AF	AG
Prescribed Dispersion m2/s	Weir Height (m)	Prescribed Reaeration /d	Bottom Algae Coverage	Bottom SOD Coverage	Prescribed SOD gO2/m2/d	Prescribed CH4 flux gC/m2/d	Prescribed NH4 flux mgN/m2/d	Prescribed Inorg P flux mgP/m2/d
0.00	0.0000							
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	3.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000
0.00	0.0000	0.000	50%	50%	0.00	0.0000	0.0000	0.0000

Figure 31. The last part of the Reach Worksheet.

Prescribed Dispersion. If the dispersion at the downstream end of a reach is known, it can be entered in column Y of the Reach sheet. If this cell is left blank, the dispersion will be automatically computed by the program.

Weir Height. If a weir is located at the reach's downstream end, this is where the weir's height is entered. If the boundary is free-flowing, a zero or blank would be entered.

Prescribed Reaeration. If the reaeration for the reach is known, it can be entered in column AA of the Reach sheet. If it is left blank, the program will internally compute it based on entries on the Rates Worksheet.

Bottom Algae Coverage. In a river, the entire bottom of a reach might not be suitable for the growth of bottom algae. Therefore, Q2K allows the user to specify the percent of the bottom where plants can grow. For example, if only one fifth of a reach's bottom area has substrate suitable for plant growth, the bottom algae coverage would be set to 20%.

Bottom SOD Coverage. In a river, the entire bottom of a reach might not be suitable for the generation of sediment oxygen demand. Therefore, Q2K allows the user to specify the percent of the bottom where sediments accumulate and SOD (along with sediment nutrient fluxes) can occur. For example, if only three quarters of a reach's bottom area has accumulated sediment mud, the bottom SOD coverage for that reach would be set to 75%.

Prescribed SOD. Q2K simulates the sediment oxygen demand for a reach as a function of the amount of detritus and phytoplankton biomass that settles from the water to the sediments at steady state. Because the sediments may also contain additional organic matter due to runoff during prior non steady-state runoff periods, Q2K allows additional SOD to be prescribed for each reach in column AD of the Reach sheet.

Prescribed CH_4 (Methane) Flux. In a similar fashion to SOD, Q2K allows an additional flux of methane (reduced carbon) to be prescribed as an input to each reach in column AE of the Reach sheet

Prescribed NH_4 (Ammonium) Flux. In a similar fashion to SOD, Q2K allows an additional flux of ammonium nitrogen to be prescribed as an input to each reach in column AF of the Reach sheet

Prescribed Inorganic Phosphorus Flux. Q2K presently does not simulate a sediment release flux of inorganic phosphorus. It does allows a flux to be prescribed as an input to each reach in column AG of the Reach sheet.

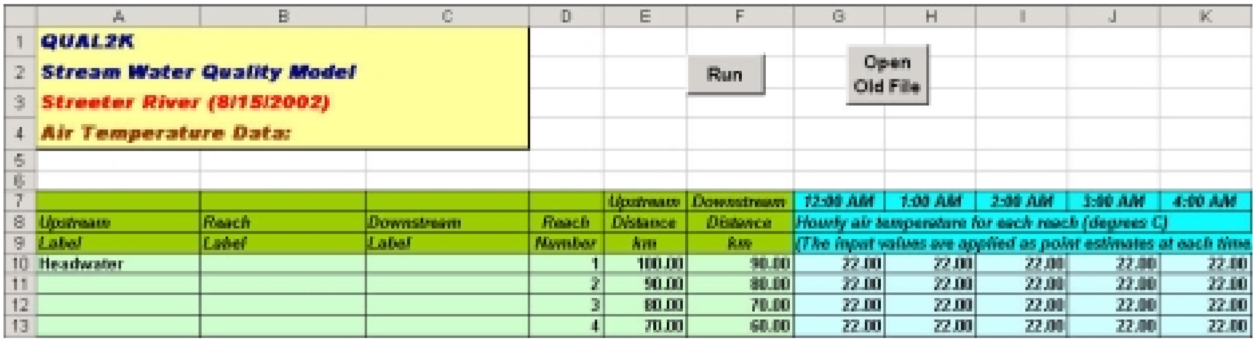
6.2.4 Meteorology and Shading Worksheets

Five worksheets are used to enter meteorological and shading data. All have the same general style as described below.

6.2.4.1 Air Temperature

This worksheet is used to enter hourly air temperatures in degrees Celcius for each of the system's reaches (Figure 32).

Labels and distances (output). The program automatically displays each reach's upstream label. reach label, downstream label, reach number, upstream distance, and downstream distance (previously entered on the Headwater and Reach Worksheets) in columns A through F.
Air Temperatures. Hourly air temperatures for each reach are entered in columns G through AD. If the values are constant over the daily cycle, just enter the mean value for all times.



QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Air Temperature Data:

Hourly air temperature for each reach (degrees C)
(The input values are applied as point estimates at each time

Upstream Label	Reach Label	Downstream Label	Reach Number	Upstream Distance km	Downstream Distance km	12:00 AM	1:00 AM	2:00 AM	3:00 AM	4:00 AM
Headwater			1	100.00	90.00	22.00	22.00	22.00	22.00	22.00
			2	90.00	80.00	22.00	22.00	22.00	22.00	22.00
			3	80.00	70.00	22.00	22.00	22.00	22.00	22.00
			4	70.00	60.00	22.00	22.00	22.00	22.00	22.00

Figure 32. The Air Temperature Worksheet.

6.2.4.2 Dew-Point Temperature

This worksheet is used to enter hourly dew-point temperatures (degrees Celcius) for each of the system's reaches.

Reach identifiers. Reach information (which was formerly entered on the Reach Worksheet) is displayed in Columns A through F.
Dew point Temperatures. Hourly dew point temperatures for each reach are entered in columns G through AD. If the values are constant over the daily cycle, just enter the mean value for all times.

6.2.4.3 Wind speed

This worksheet is used to enter hourly wind speeds (meters per second) for each of the system's reaches.

Reach identifiers. Reach information (which was formerly entered on the Reach Worksheet) is displayed in Columns A through F.

Dew point Temperatures. Hourly wind speeds for each reach are entered in columns G through AD. If the values are constant over the daily cycle, just enter the mean value for all times.

6.2.4.4 Cloud cover

This worksheet is used to enter hourly cloud cover (% of sky covered) for each of the system's reaches.

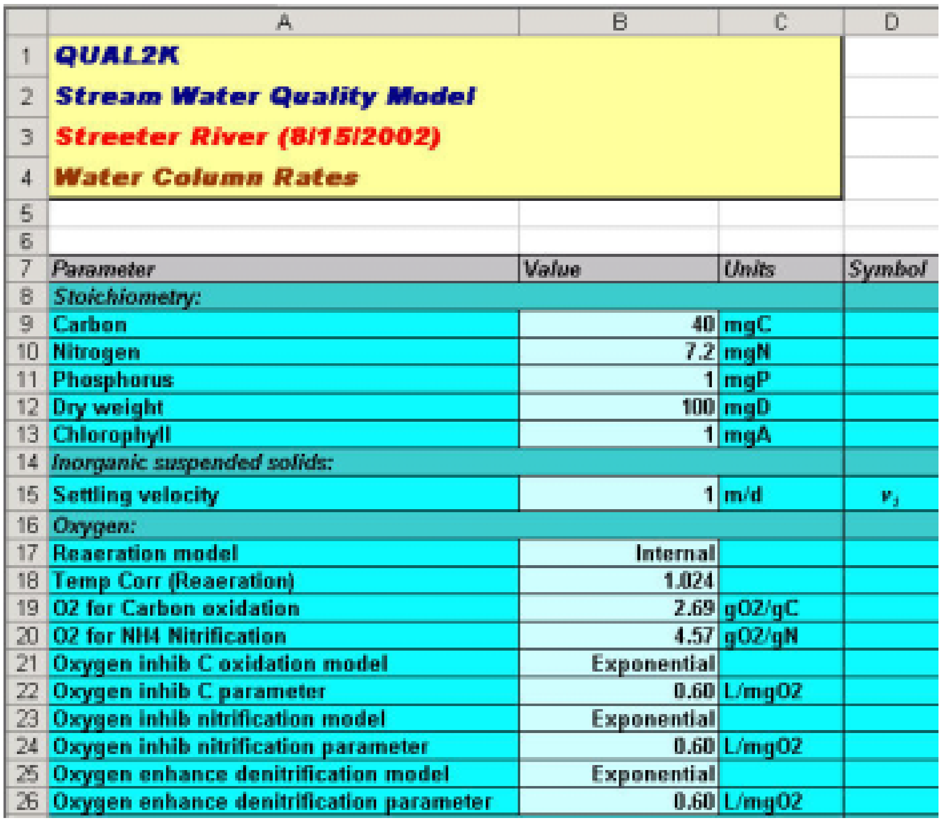
Reach identifiers. Reach information (which was formerly entered on the Reach Worksheet) is displayed in Columns A through F.
Dew point Temperatures. Hourly cloud cover for each reach are entered in columns G through AD. If the values are constant over the daily cycle, just enter the mean value for all times.

6.2.4.5 Shade

This worksheet is used to enter hourly shading for each of the system's reaches. Shading is defined as the fraction of solar radiation that is blocked because of shade from topography and vegetation.

6.2.5 Rates Worksheet

This worksheet is used to enter the model's rate parameters (Figure 33 through Figure 36).



QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Water Column Rates

Parameter	Value	Units	Symbol
Stoichiometry:			
Carbon	40	mgC	
Nitrogen	7.2	mgN	
Phosphorus	1	mgP	
Dry weight	100	mgD	
Chlorophyll	1	mgA	
Inorganic suspended solids:			
Settling velocity	1	m/d	v_i
Oxygen:			
Reaeration model	Internal		
Temp Corr (Reaeration)	1.024		
O2 for Carbon oxidation	2.69	gO2/gC	
O2 for NH4 Nitrification	4.57	gO2/gN	
Oxygen inhib C oxidation model	Exponential		
Oxygen inhib C parameter	0.60	L/mgO2	
Oxygen inhib nitrification model	Exponential		
Oxygen inhib nitrification parameter	0.60	L/mgO2	
Oxygen enhance denitrification model	Exponential		
Oxygen enhance denitrification parameter	0.60	L/mgO2	

Figure 33. The part of the Rates Worksheet used to input stoichiometry and rate parameters for inorganic suspended solids and oxygen.

Stoichiometry:

The model assumes a fixed stoichiometry of plant and detrital matter. Recommended values for these parameters are listed in Table 8.

Table 8 Recommended values for stoichiometry.

Carbon	**40 mgC**
Nitrogen	**7.2 mgN**
Phosphorus	**1 mgP**
Dry weight	**100 mgD**
Chlorophyll	**1 mgA**

It should be noted that chlorophyll is the most variable of these values with a range from about 0.5 to 2 mgA.

.

Inorganic suspended solids:

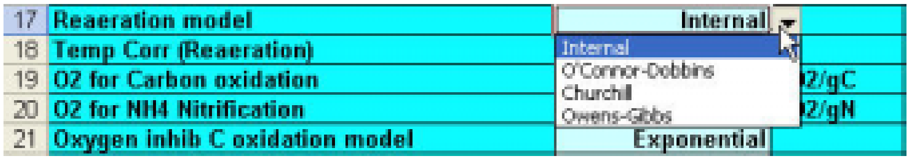
Settling velocity

Oxygen:

Reaeration model. Recall that the Reach Worksheet (Figure 31) can be used to specify the reaeration rate for each reach. Note that when the reaeration is entered this way, all other options are overridden. If reaeration is not specified on the Reach sheet, a pull-down menu (Figure 34) is used to select among several options to determine river reaeration:

- **Internal.** The reaeration is computed internally depending on the river's depth and velocity (Covar 1976)
- **O'Connor-Dobbins formula.**
- **Churchill formula.**
- **Owens-Gibbs formula.**

The selected option will then be applied to all the unspecified cells.



17	Reaeration model	Internal	
18	Temp Corr (Reaeration)		
19	O2 for Carbon oxidation		2/gC
20	O2 for NH4 Nitrification		2/gN
21	Oxygen inhib C oxidation model	Exponential	

Figure 34. The pull-down menu for global reaeration rates.

Temperature correction (reaeration). Suggested value: 1.024.
O_2 for CBOD oxidation. Suggested value: 2.69 gO_2/gC.
O_2 for NH_4 nitrification. Suggested value: 4.57 gO_2/gC.
Oxygen inhibition C oxidation model. A pull-down menus is used to choose among the following options:

- Half-saturation
- Exponential
- Second order

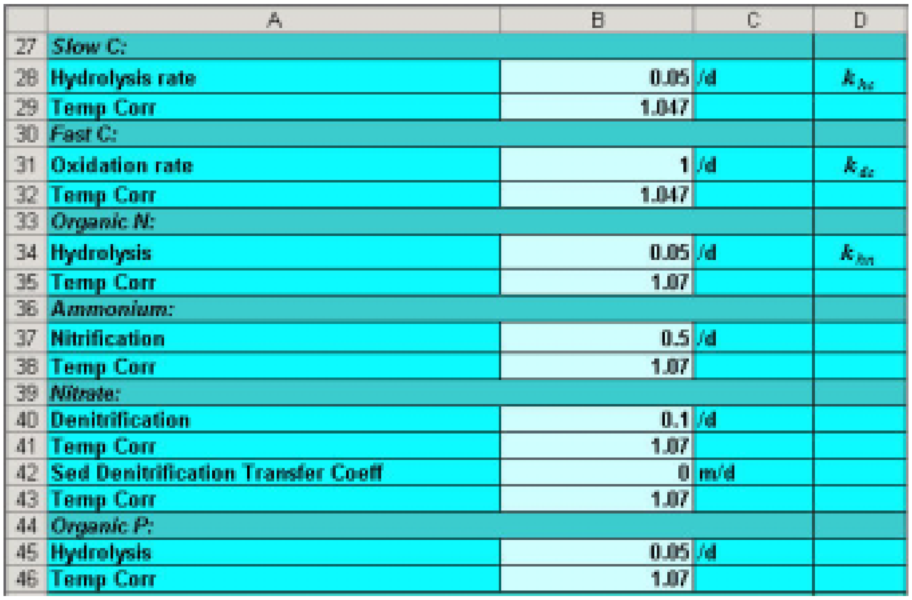
Oxygen inhibition C parameter. This should be the proper parameter for the chosen oxygen inhibition model specified in cell B21.
Oxygen inhibition nitrification model. A pull-down menus is used to choose among the following options:

- Half-saturation
- Exponential
- Second order

Oxygen inhibition nitrification parameter. This should be the proper parameter for the chosen oxygen inhibition model specified in cell B23.
Oxygen enhancement denitrification model. A pull-down menus is used to choose among the following options:

- Half-saturation
- Exponential
- Second order

Oxygen enhancement denitrification parameter. This should be the proper parameter for the oxygen enhancement model specified in cell specified in cell B25.



	A	B	C	D
27	*Slow C:*			
28	Hydrolysis rate	0.05	/d	k_{hc}
29	Temp Corr	1.047		
30	*Fast C:*			
31	Oxidation rate	1	/d	k_{dc}
32	Temp Corr	1.047		
33	*Organic N:*			
34	Hydrolysis	0.05	/d	k_{hn}
35	Temp Corr	1.07		
36	*Ammonium:*			
37	Nitrification	0.5	/d	
38	Temp Corr	1.07		
39	*Nitrate:*			
40	Denitrification	0.1	/d	
41	Temp Corr	1.07		
42	Sed Denitrification Transfer Coeff	0	m/d	
43	Temp Corr	1.07		
44	*Organic P:*			
45	Hydrolysis	0.05	/d	
46	Temp Corr	1.07		

Figure 35. The part of the Rates Worksheet used to input rate parameters for slow CBOD, fast CBOD, organic N, ammonium, nitrate, and organic P.

Slow C:

Hydrolysis rate
Temperature correction

Fast C:

Oxidation rate
Temperature correction

Organic N:

Hydrolysis
Temperature correction

Ammonium:

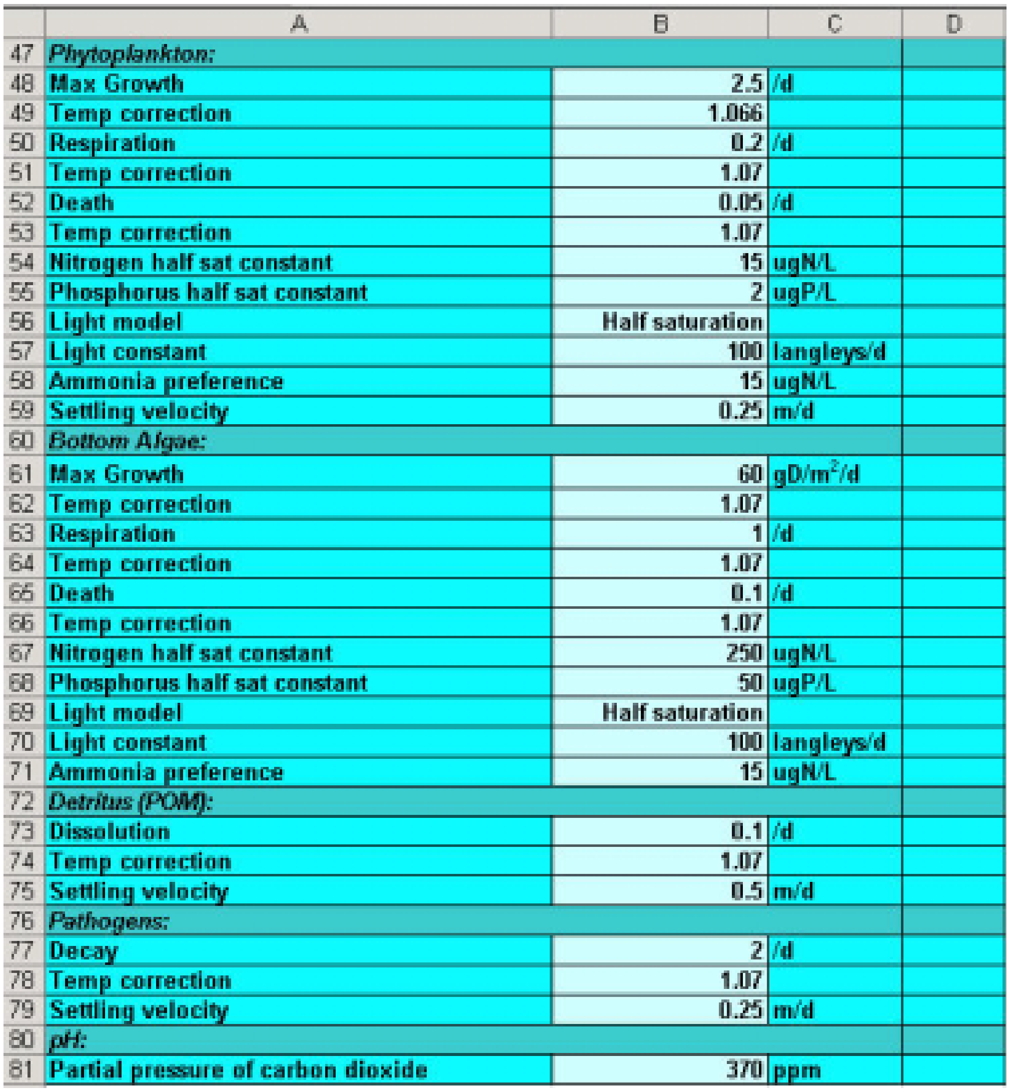
Nitrification
Temperature correction

Nitrate:

Denitrification
Temperature correction
Sediment denitrification transfer coefficient. This is the velocity at which nitrate diffuses into the sediments where it is denitrified to nitrogen gas.
Temperature correction

Organic P:

Hydrolysis
Temperature correction



	A	B	C	D
47	*Phytoplankton:*			
48	Max Growth	2.5	/d	
49	Temp correction	1.066		
50	Respiration	0.2	/d	
51	Temp correction	1.07		
52	Death	0.05	/d	
53	Temp correction	1.07		
54	Nitrogen half sat constant	15	ugN/L	
55	Phosphorus half sat constant	2	ugP/L	
56	Light model	Half saturation		
57	Light constant	100	langleys/d	
58	Ammonia preference	15	ugN/L	
59	Settling velocity	0.25	m/d	
60	*Bottom Algae:*			
61	Max Growth	60	gD/m^2/d	
62	Temp correction	1.07		
63	Respiration	1	/d	
64	Temp correction	1.07		
65	Death	0.1	/d	
66	Temp correction	1.07		
67	Nitrogen half sat constant	250	ugN/L	
68	Phosphorus half sat constant	50	ugP/L	
69	Light model	Half saturation		
70	Light constant	100	langleys/d	
71	Ammonia preference	15	ugN/L	
72	*Detritus (POM):*			
73	Dissolution	0.1	/d	
74	Temp correction	1.07		
75	Settling velocity	0.5	m/d	
76	*Pathogens:*			
77	Decay	2	/d	
78	Temp correction	1.07		
79	Settling velocity	0.25	m/d	
80	*pH:*			
81	Partial pressure of carbon dioxide	370	ppm	

Figure 36. The part of the Rates Worksheet used to input rate parameters for phytoplankton, bottom algae, detritus, pathogens and pH.

Floating Plants (Phytoplankton):

Maximum Growth Rate
Temperature correction
Respiration
Temperature correction
Death
Temperature correction
Nitrogen half saturation constant
Phosphorus half saturation constant
Light model. A pull-down menu (Figure 37) is used to select among three light models:

Figure 37. The pull-down menu to select the light model for phytoplankton growth.

Light constant
Ammonia preference
Settling velocity

Bottom algae:

Maximum growth
Temperature correction
Respiration
Temperature correction
Death
Temperature correction
Nitrogen half saturation constant
Phosphorus half saturation constant
Light model. A pull-down menu identical to Figure 37 is used to select among three light models: Half saturation, Smith and Steele.
Light Constant
Ammonia preference

Detritus (POM):

Dissolution
Temperature correction
Settling Velocity

Pathogens:

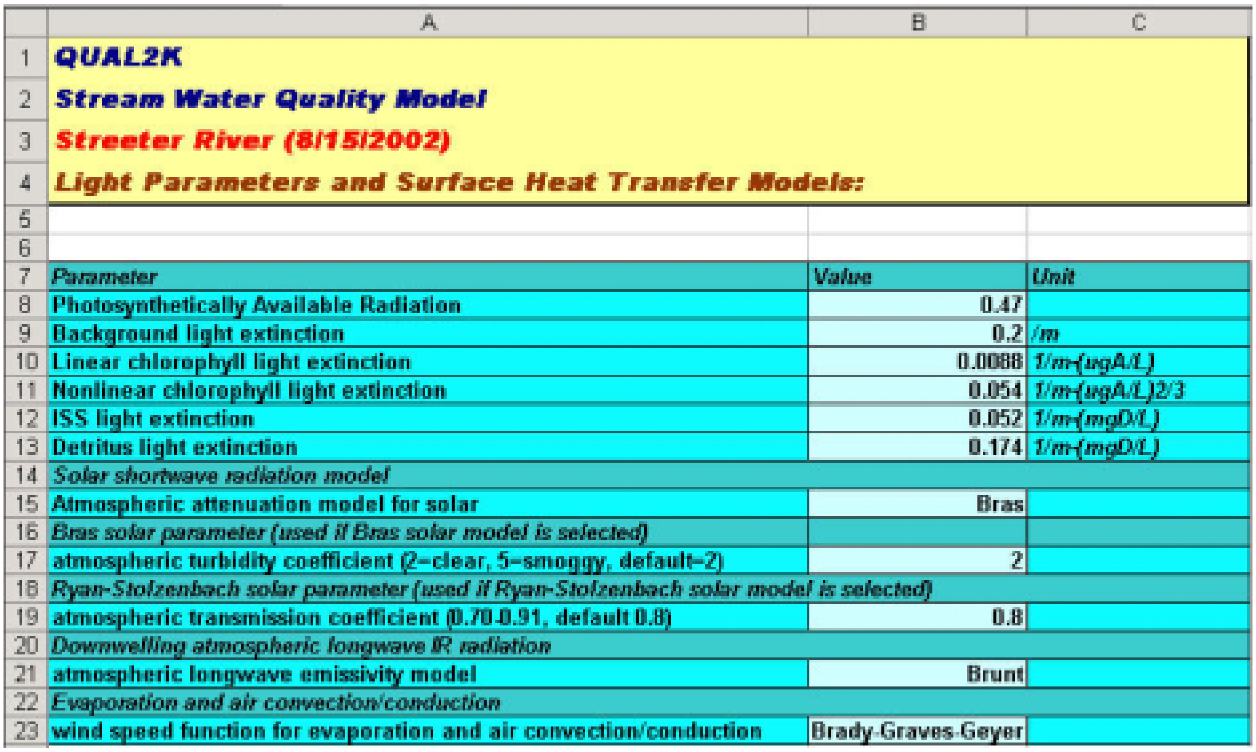
Decay
Temperature correction
Settling Velocity

pH:

pCO2. The partial pressure of CO_2 in the atmosphere (see Figure 20 for suggested values).

6.2.6 Light and Heat Worksheet

This worksheet is used to enter information related the system's light and heat parameters.



	A	B	C
1	QUAL2K		
2	Stream Water Quality Model		
3	Streeter River (8/15/2002)		
4	Light Parameters and Surface Heat Transfer Models:		
5			
6			
7	Parameter	Value	Unit
8	Photosynthetically Available Radiation	0.47	
9	Background light extinction	0.2	/m
10	Linear chlorophyll light extinction	0.0088	1/m-(ugA/L)
11	Nonlinear chlorophyll light extinction	0.054	1/m-(ugA/L)2/3
12	ISS light extinction	0.052	1/m-(mgD/L)
13	Detritus light extinction	0.174	1/m-(mgD/L)
14	Solar shortwave radiation model		
15	Atmospheric attenuation model for solar	Bras	
16	Bras solar parameter (used if Bras solar model is selected)		
17	atmospheric turbidity coefficient (2=clear, 5=smoggy, default=2)	2	
18	Ryan-Stolzenbach solar parameter (used if Ryan-Stolzenbach solar model is selected)		
19	atmospheric transmission coefficient (0.70-0.91, default 0.8)	0.8	
20	Downwelling atmospheric longwave IR radiation		
21	atmospheric longwave emissivity model	Brunt	
22	Evaporation and air convection/conduction		
23	wind speed function for evaporation and air convection/conduction	Brady-Graves-Geyer	

Figure 38. The Light Worksheet used to input light-related parameters.

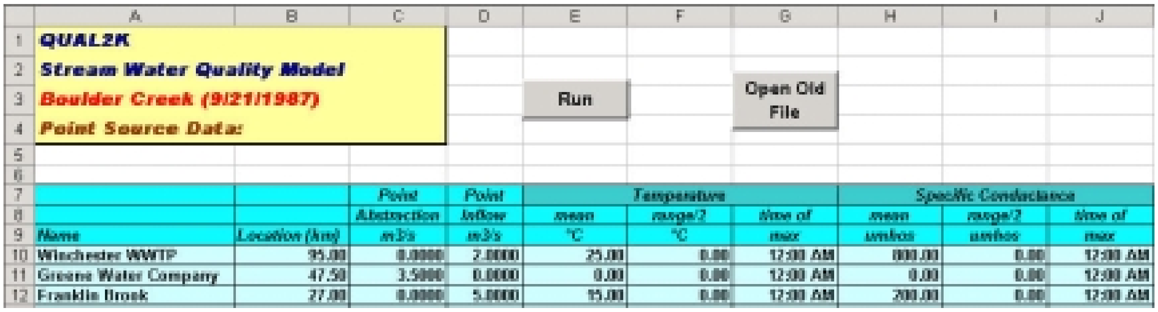
Photosynthetically Available Radiation. This is the fraction of incoming solar radiation that is available for photosynthesis. It is recommended that this value be set to 0.47.
Background light extinction. This parameter accounts for light extinction due to water and color.
Linear chlorophyll light extinction. This parameter accounts for the linear dependence of light extinction due to phytoplankton chlorophyll *a*. According to Riley (1956), this parameter should be set to 0.0088/(m µgA/L).
Nonlinear chlorophyll light extinction. This parameter accounts for the nonlinear dependence of light extinction due to phytoplankton chlorophyll *a*. According to Riley (1956), this parameter should be set to 0.054/(m (µgA/L)$^{2/3}$). Note that if the relationship is believed to be linear, this parameter can be set to zero and the linear coefficient modified accordingly.
Inorganic suspended solids light extinction. This parameter accounts for the nonlinear dependence of light extinction on inorganic suspended solids.
Detritus light extinction. This parameter accounts for the nonlinear dependence of light extinction on detritus.
Atmospheric attenuation model for solar (default: Bras). A pull down menu allows you to choose among 2 options: the Bras or the Ryan-Stolzenbach models.
Atmospheric turbidity coefficient (2=clear, 5=smoggy, default=2). This is used if the Bras solar model is selected
Atmospheric transmission coefficient (0.70-0.91, default 0.8). This is used if the Ryan-Stolzenbach solar model is selected.

Atmospheric longwave emissivity model (default: Brutsaert). A pull down menu allows you to choose among 3 options: the Brutsaert, Brunt or Koberg models.

Wind speed function for evaporation and air convection/conduction (default: Brady-Graves-Geyer). A pull down menu allows you to choose among 3 options: the Brady-Graves-Geyer, the Adams 1, or the Adams 2 models.

6.2.7 Point Sources Worksheet

This worksheet is used to enter information related the system’s point sources.



QUAL2K
Stream Water Quality Model
Boulder Creek (9/21/1987)
Point Source Data:

		Point	Point	Temperature			Specific Conductance		
		Abstraction	Inflow	mean	range/2	time of	mean	range/2	time of
Name	Location (km)	m3/s	m3/s	°C	°C	max	umhos	umhos	max
Winchester WWTP	95.00	0.0000	2.0000	25.00	0.00	12:00 AM	800.00	0.00	12:00 AM
Greene Water Company	47.50	3.5000	0.0000	0.00	0.00	12:00 AM	0.00	0.00	12:00 AM
Franklin Brook	27.00	0.0000	5.0000	15.00	0.00	12:00 AM	200.00	0.00	12:00 AM

Figure 39. The Point Sources Worksheet.

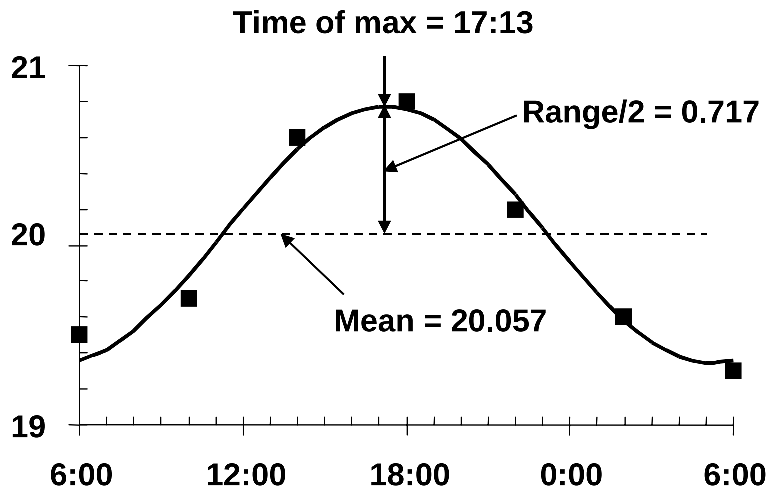
Name. User-specified label to identify the particular diffuse abstraction of inflow.
Location. The kilometer where the point source or abstraction enters or leaves the river.
Source Inflows and Outflows. A source can either be an inflow (loading or tributary) or an outflow (abstraction). Note that it can not be both. If there is an abstraction flow (i.e., a positive value in column C), the remaining information in columns D through T will be ignored.

Point abstraction. For an abstraction, a positive[4] value for flow (m^3/s) must be entered. If this is done, the values in columns D through T should be left blank.

Point inflow. For an input, a value for flow (m^3/s) must be entered in column D. Column C should be a zero or a blank.

Constituents. The temperature and the water quality concentrations of the inflow are entered in columns E through AZ.

QUAL2K allows the temperature and concentrations of each point source to be entered as a sinusoid that varies over the diel cycle. Figure 40 shows an example for the temperature of the Boulder CO WWTP.



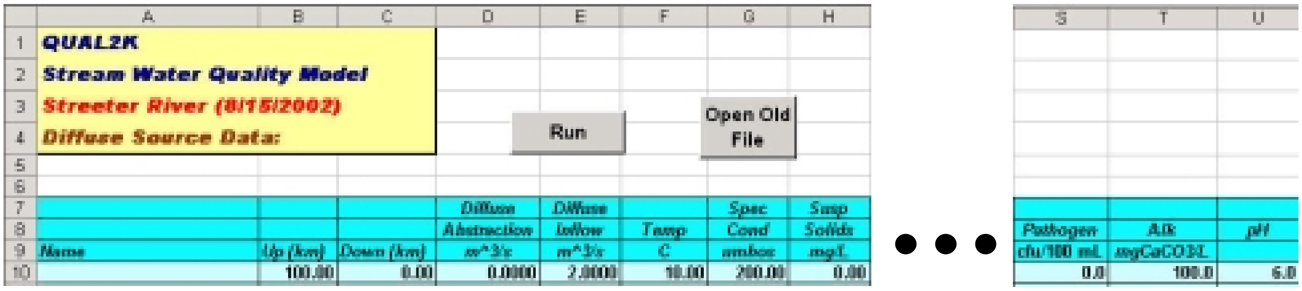
---

[4] Some software treats an abstraction as a negative inflow. In Q2K, the flow is entered as a positive number and the software internally calculates it as a loss from the reach.

Figure 40. Temperature for the Boulder, CO wastewater treatment plant effluent on Sept. 21-22, 1987 along with a sinusoidal fit to the data.

6.2.8 Diffuse Sources Worksheet

This worksheet is used to enter information related the system's diffuse (i.e., non-point) sources.



QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Diffuse Source Data:

Name	Up (km)	Down (km)	Diffuse Abstraction m^3/s	Diffuse Inflow m^3/s	Temp C	Spec Cond umhos	Susp Solids mg/L	Pathogen cfu/100 mL	Alk mgCaCO3/L	pH
	100.00	0.00	0.0000	2.0000	10.00	200.00	0.00	0.0	100.0	6.0

Figure 41. The Diffuse Sources Worksheet.

Name. User-specified label to identify the particular diffuse abstraction of inflow.
Location. The upstream and downstream kilometers over which the diffuse source or abstraction enters or leaves the river.
Source inflows and outflows. A distributed source can either be an inflow (loading or tributary) or an outflow (abstraction). Note that it can not be both. If there is an abstraction flow (i.e., a positive value in column D), the remaining information in columns E through U will be ignored.

Diffuse abstraction. For an abstraction, a positive[5] value for flow (m^3/s) must be entered. If this is done, the values in columns U through U should be left blank.

Diffuse inflow. For an input, a value for flow (m^3/s) must be entered in column E. Column D should be a zero or a blank.

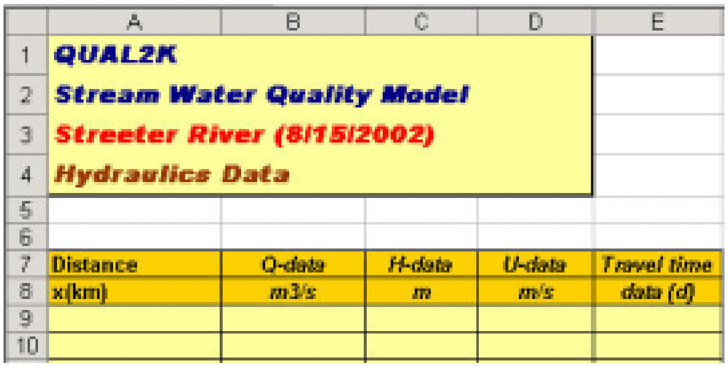
Constituents. The temperature and the water quality concentrations of the diffuse inflow are entered in columns F through U.

---

[5] Some software treats an abstraction as a negative inflow. In Q2K, the flow is entered as a positive number and the software internally calculates it as a loss from the reach.

6.2.9 Hydraulics Data Worksheet

This worksheet is used to enter data related to the system's hydraulics.



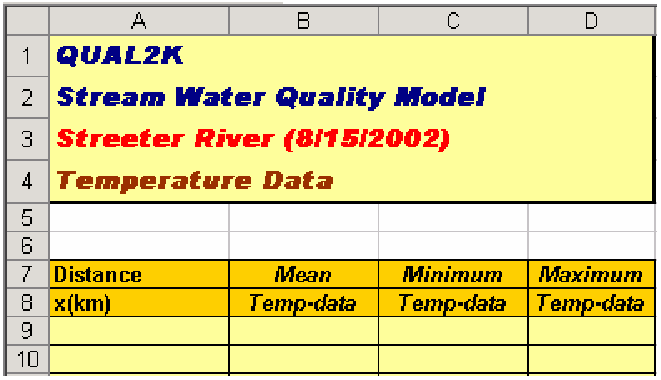
	A	B	C	D	E
1	QUAL2K				
2	Stream Water Quality Model				
3	Streeter River (8/15/2002)				
4	Hydraulics Data				
5					
6					
7	Distance	Q-data	H-data	U-data	Travel time
8	x(km)	m3/s	m	m/s	data (d)
9					
10					

Figure 42. The Hydraulics Data Worksheet.

Distance. This is the distance (km) at which the hydraulics data are plotted.
***Q*-data.** Flow data in m^3/s.
***H*-data.** Depth data in m.
***U*-data.** Velocity data in m/s.
Travel time-data. Travel time in days.

6.2.10 Temperature Data Worksheet

This worksheet is used to enter temperature data.



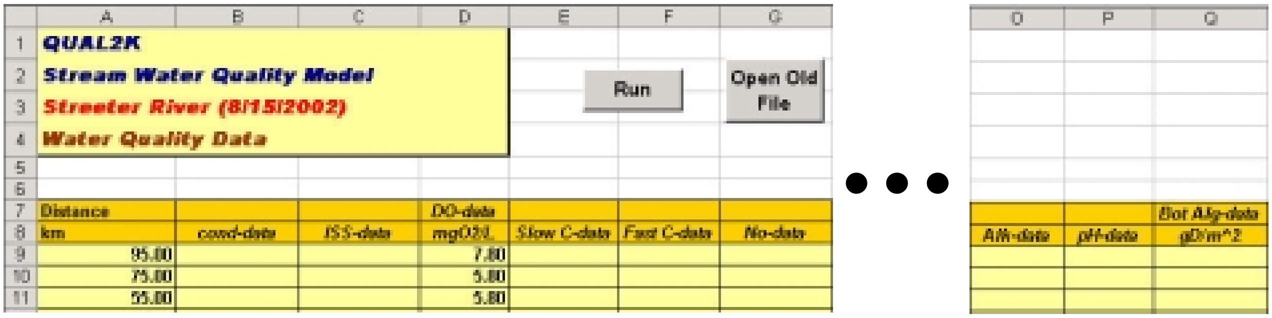
	A	B	C	D
1	QUAL2K			
2	Stream Water Quality Model			
3	Streeter River (8/15/2002)			
4	Temperature Data			
5				
6				
7	Distance	Mean	Minimum	Maximum
8	x(km)	Temp-data	Temp-data	Temp-data
9				
10				

Figure 43. The Temperature Data Worksheet.

Distance. This is the distance (km) at which the temperature data are plotted.
Mean Temperature-data. The mean temperature in °C.
Minimum Temperature-data. The minimum temperature in °C.
Maximum Temperature-data. The maximum temperature in °C.

6.2.11 WQ Data Worksheet

This worksheet is used to enter mean daily values for water quality data. The first part of the worksheet is shown in Figure 44.



QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Water Quality Data

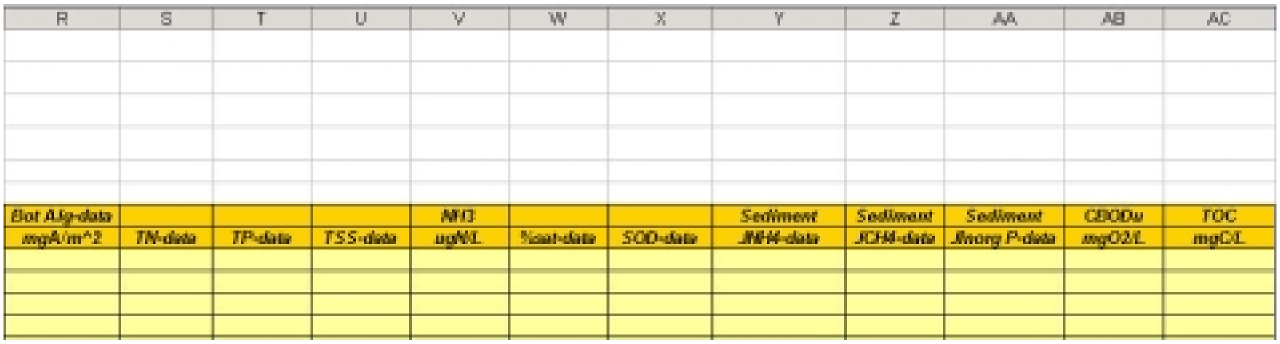
	A	B	C	D	E	F	G
7	Distance			DO-data			
8	km	cond-data	ISS-data	mgO2/L	Slow C-data	Fast C-data	No-data
9	95.00			7.80			
10	75.00			5.80			
11	55.00			5.80			

O	P	Q
		Bot Alg-data
Alk-data	pH-data	gD/m^2

Figure 44. The first part of the Water Quality Data Worksheet.

Distance. This is the distance (km) at which the water quality data are plotted.
Constituents. The mean measured water quality concentrations are entered in columns B through R.
Other Concentrations and Fluxes. A variety of other concentrations and fluxes are entered in Columns Q through AC as shown in Figure 45. These are

Bottom Algae.
Total nitrogen-data.
Total phosphorus-data.
Total suspended solids-data.
NH_3 (unionized ammonia)-data
% saturation-data.
SOD-data
Sediment ammonium flux.
Sediment methane flux.
Sediment inorganic phosphorus flux.
Ultimate carbonaceous BOD. This is the total of detritus, slow CBOD, fast CBOD, and phytoplankton biomass expressed as oxygen equivalents.
Total Organic Carbon. This is the total of inorganic suspended solids, phytoplankton biomass and detritus expressed as dry weight.



R	S	T	U	V	W	X	Y	Z	AA	AB	AC
Bot Alg-data				NH3			Sediment	Sediment	Sediment	CBODu	TOC
mgA/m^2	TN-data	TP-data	TSS-data	ugN/L	%sat-data	SOD-data	JNH4-data	JCH4-data	Jinorg P-data	mgO2/L	mgC/L

Figure 45. The last part of the Water Quality Data Worksheet.

6.2.12 WQ Data Min Worksheet

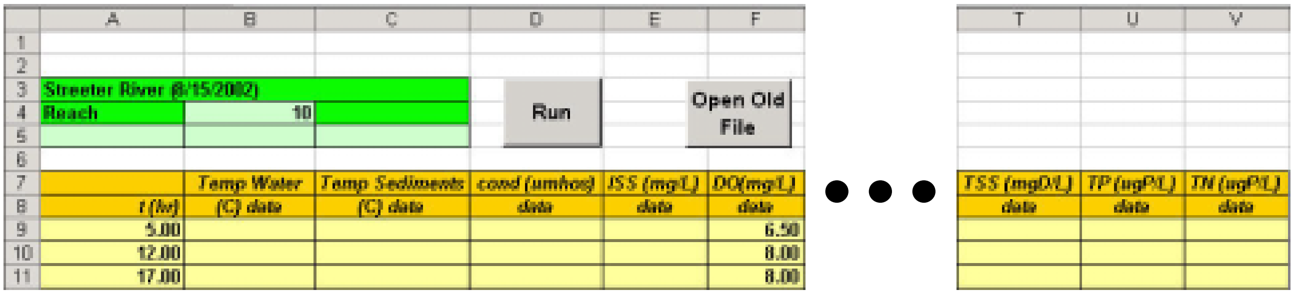
This worksheet is used to enter minimum daily values for water quality data. The layout is the same as for the WQ Data Worksheet.

6.2.13 WQ Data Max Worksheet

This worksheet is used to enter maximum daily values for water quality data. The layout is the same as for the WQ Data Worksheet.

6.2.14 Diel Data Worksheet

This worksheet is used to enter diel data for a selected reach. This data is then plotted as points on the graphs of diel model output.



Streeter River (8/15/2002)

Reach 10

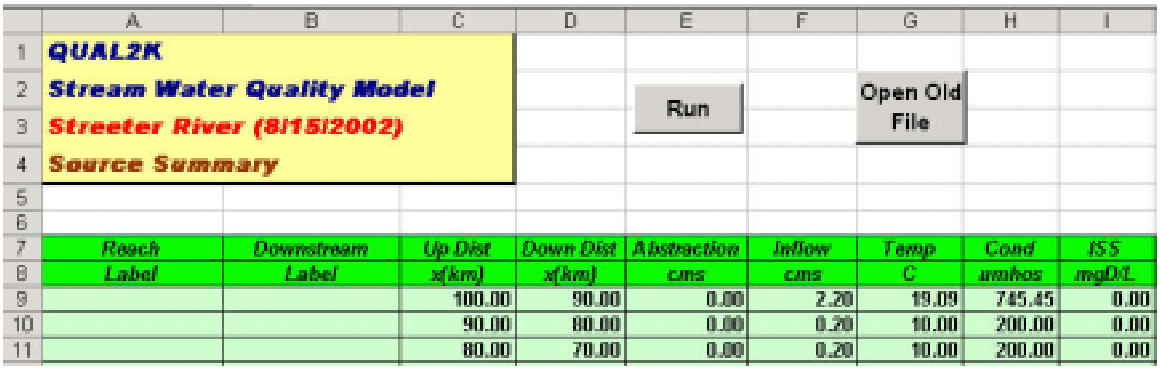
t (hr)	Temp Water (C) data	Temp Sediments (C) data	cond (umhos) data	ISS (mg/L) data	DO(mg/L) data	TSS (mgD/L) data	TP (ugP/L) data	TN (ugP/L) data
5.00					6.50			
12.00					8.00			
17.00					8.00			

Figure 46. The Diel Data Worksheet.

6.2.15 Output Worksheets

These are a series of worksheets that present tables of numerical output generated by Q2K.

Source Summary. This worksheet summarizes the total loading for each model reach.

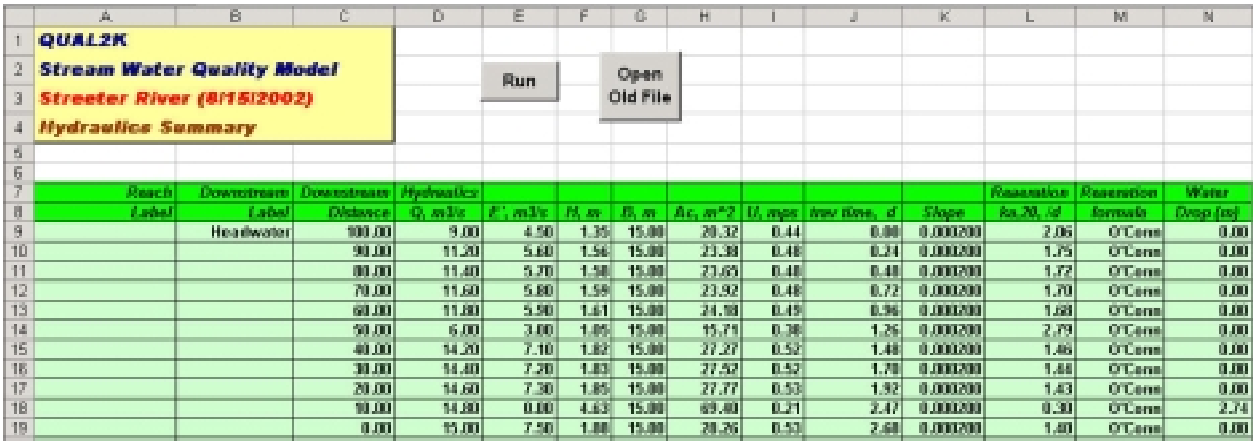


QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Source Summary

Reach Label	Downstream Label	Up Dist x(km)	Down Dist x(km)	Abstraction cms	Inflow cms	Temp C	Cond umhos	ISS mgD/L
		100.00	90.00	0.00	2.20	19.09	745.45	0.00
		90.00	80.00	0.00	0.20	10.00	200.00	0.00
		80.00	70.00	0.00	0.20	10.00	200.00	0.00

Figure 47. The Source Summary Worksheet.

Hydraulics Summary. This worksheet summarizes the hydraulic parameters for each model reach.

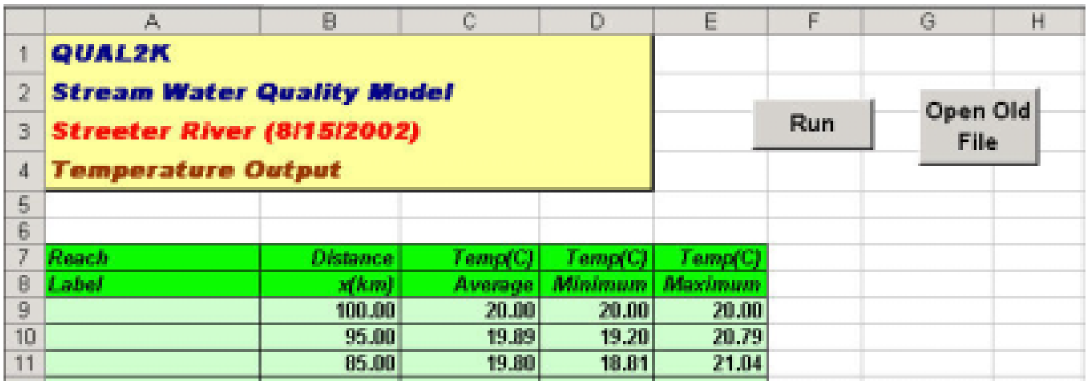


QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Hydraulics Summary

Reach Label	Downstream Label	Downstream Distance	Hydraulics Q, m3/s	E', m3/s	H, m	B, m	Ac, m^2	U, mps	trav time, d	Slope	Reaeration ka,20, /d	Reaeration formula	Water Drop (m)
	Headwater	100.00	9.00	4.50	1.35	15.00	20.32	0.44	0.00	0.000200	2.06	O'Conn	0.00
		90.00	11.20	5.60	1.56	15.00	23.38	0.48	0.24	0.000200	1.75	O'Conn	0.00
		80.00	11.40	5.70	1.58	15.00	23.65	0.48	0.48	0.000200	1.72	O'Conn	0.00
		70.00	11.60	5.80	1.59	15.00	23.92	0.48	0.72	0.000200	1.70	O'Conn	0.00
		60.00	11.80	5.90	1.61	15.00	24.18	0.49	0.96	0.000200	1.68	O'Conn	0.00
		50.00	6.00	3.00	1.05	15.00	15.71	0.38	1.26	0.000200	2.79	O'Conn	0.00
		40.00	14.20	7.10	1.82	15.00	27.27	0.52	1.48	0.000200	1.46	O'Conn	0.00
		30.00	14.40	7.20	1.83	15.00	27.52	0.52	1.70	0.000200	1.44	O'Conn	0.00
		20.00	14.60	7.30	1.85	15.00	27.77	0.53	1.92	0.000200	1.43	O'Conn	0.00
		10.00	14.80	0.00	4.63	15.00	69.40	0.21	2.47	0.000200	0.30	O'Conn	2.74
		0.00	15.00	7.50	1.88	15.00	28.26	0.53	2.68	0.000200	1.40	O'Conn	0.00

Figure 48. The Hydraulics Summary Worksheet.

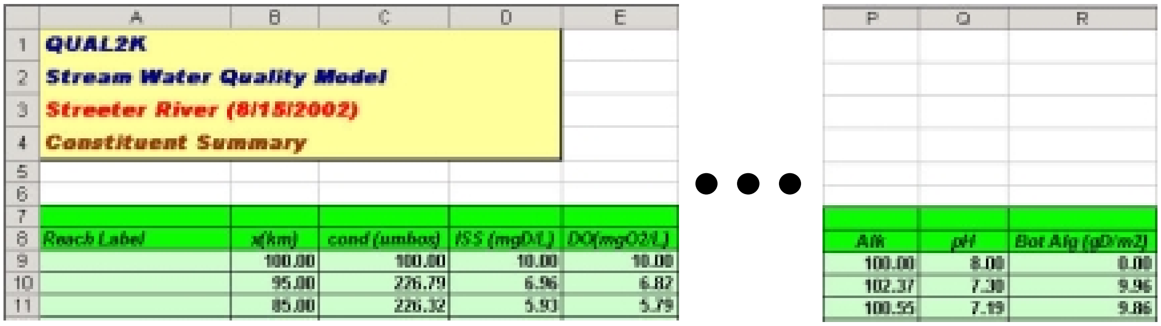
Temperature Output. This worksheet summarizes the temperature output for each model reach.


QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Temperature Output

Reach Label	Distance x(km)	Temp(C) Average	Temp(C) Minimum	Temp(C) Maximum
	100.00	20.00	20.00	20.00
	95.00	19.89	19.20	20.79
	85.00	19.80	18.81	21.04

Figure 49. The Temperature Output Worksheet.

WQ Output. This worksheet summarizes the mean concentration output for each model reach.


QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Constituent Summary

Reach Label	x(km)	cond (umhos)	ISS (mgD/L)	DO(mgO2/L)
	100.00	100.00	10.00	10.00
	95.00	226.79	6.96	6.82
	85.00	226.32	5.93	5.79

Alk	pH	Bot Alg (gD/m2)
100.00	8.00	0.00
102.37	7.30	9.96
100.55	7.19	9.86

Figure 50. The first part of the Water Quality Output Worksheet.

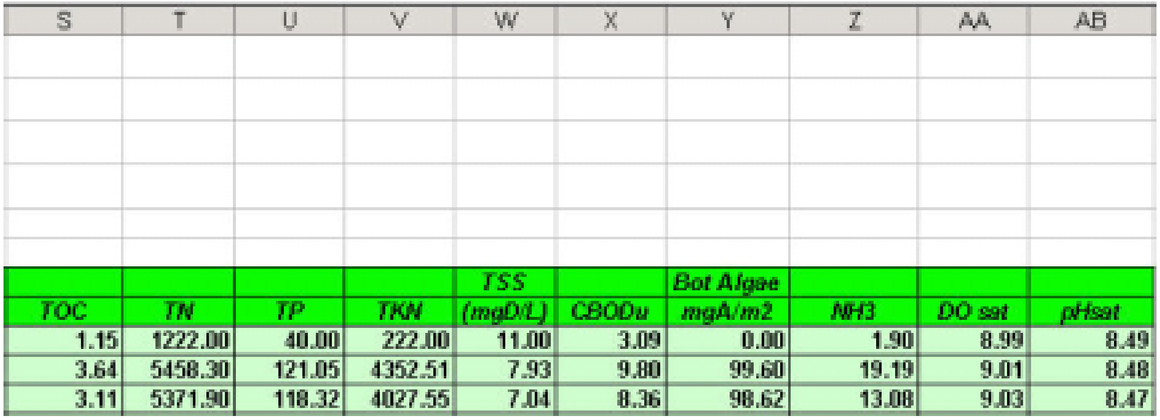
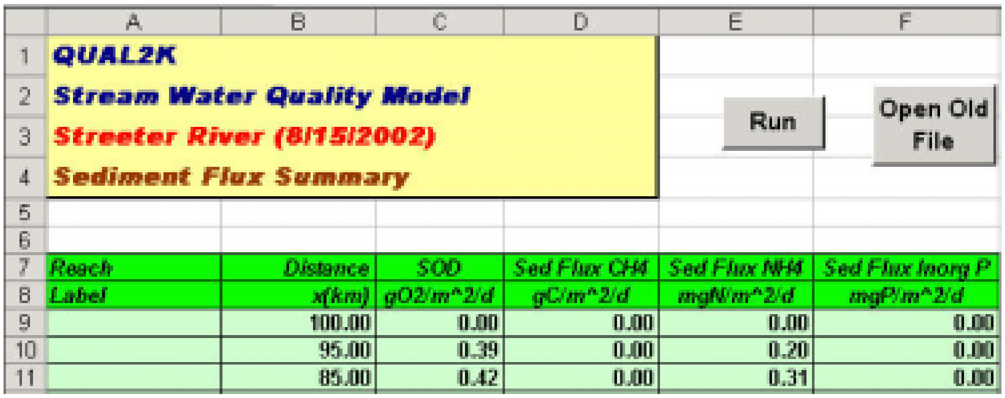

TOC	TN	TP	TKN	TSS (mgD/L)	CBODu	Bot Algae mgA/m2	NH3	DO sat	pHsat
1.15	1222.00	40.00	222.00	11.00	3.09	0.00	1.90	8.99	8.49
3.64	5458.30	121.05	4352.51	7.93	9.80	99.60	19.19	9.01	8.48
3.11	5371.90	118.32	4027.55	7.04	8.36	98.62	13.08	9.03	8.47

Figure 51. The last part of the Water Quality Output Worksheet.

WQ Min. This worksheet summarizes the minimum concentration output for each model reach.

WQ Max. This worksheet summarizes the maximum concentration output for each model reach.

Sediment Fluxes. This worksheet summarizes the fluxes of oxygen and nutrients between the water and the underlying sediment compartment for each model reach.

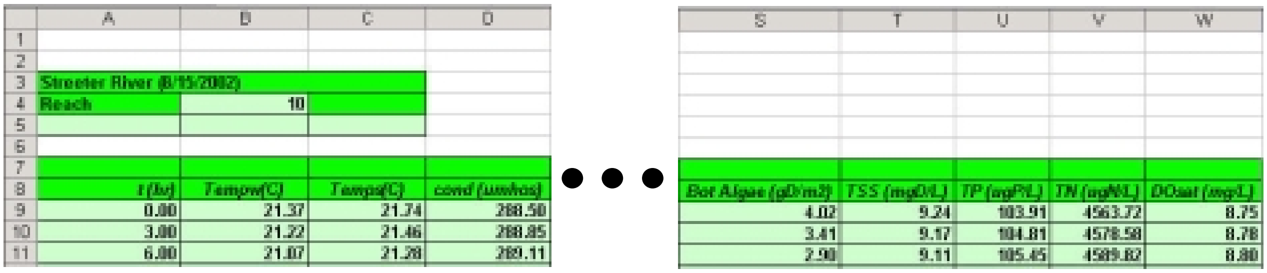


QUAL2K
Stream Water Quality Model
Streeter River (8/15/2002)
Sediment Flux Summary

Reach Label	Distance x(km)	SOD gO2/m^2/d	Sed Flux CH4 gC/m^2/d	Sed Flux NH4 mgN/m^2/d	Sed Flux Inorg P mgP/m^2/d
	100.00	0.00	0.00	0.00	0.00
	95.00	0.39	0.00	0.20	0.00
	85.00	0.42	0.00	0.31	0.00

Figure 52. The Sediment Flux Worksheet.

Diel Output Worksheet. This worksheet displays diel output for temperature and water quality constituent data. Note that along with the water temperature, the temperature of the surface sediments are also displayed. In addition, the diel variation in total suspended solids, total phosphorus, total nitrogen, and oxygen saturation are also displayed.



Streeter River (8/15/2002)
Reach 10

t (hr)	Tempw(C)	Temps(C)	cond (umhos)	Bot Algae (gD/m2)	TSS (mgD/L)	TP (ugP/L)	TN (ugN/L)	DOsat (mg/L)
0.00	21.37	21.74	288.50	4.02	9.24	103.91	4563.72	8.75
3.00	21.22	21.46	288.85	3.41	9.17	104.81	4578.58	8.78
6.00	21.07	21.28	289.11	2.90	9.11	105.45	4589.82	8.80

Figure 53. The Diel Output Worksheet.

6.2.16 Spatial Charts

QUAL2K displays a series of charts that plot the model output and data versus distance (km) along the river.

Figure 54 shows an example of the plot for dissolved oxygen. The black line is the simulated mean DO (as displayed on the WQ Worksheet), whereas the dashed red lines are the minimum (WQ Min Worksheet) and maximum (WQ Max Worksheet) values, respectively. The black squares are the measured mean data points that were entered on the WQ Data Worksheet. The white squares are the minimum (WQ Min Worksheet) and maximum (WQ Max Worksheet) data points, respectively. The plot is labeled with the river name and the simulation date. Notice that this plot also displays the oxygen saturation as a dashed line.

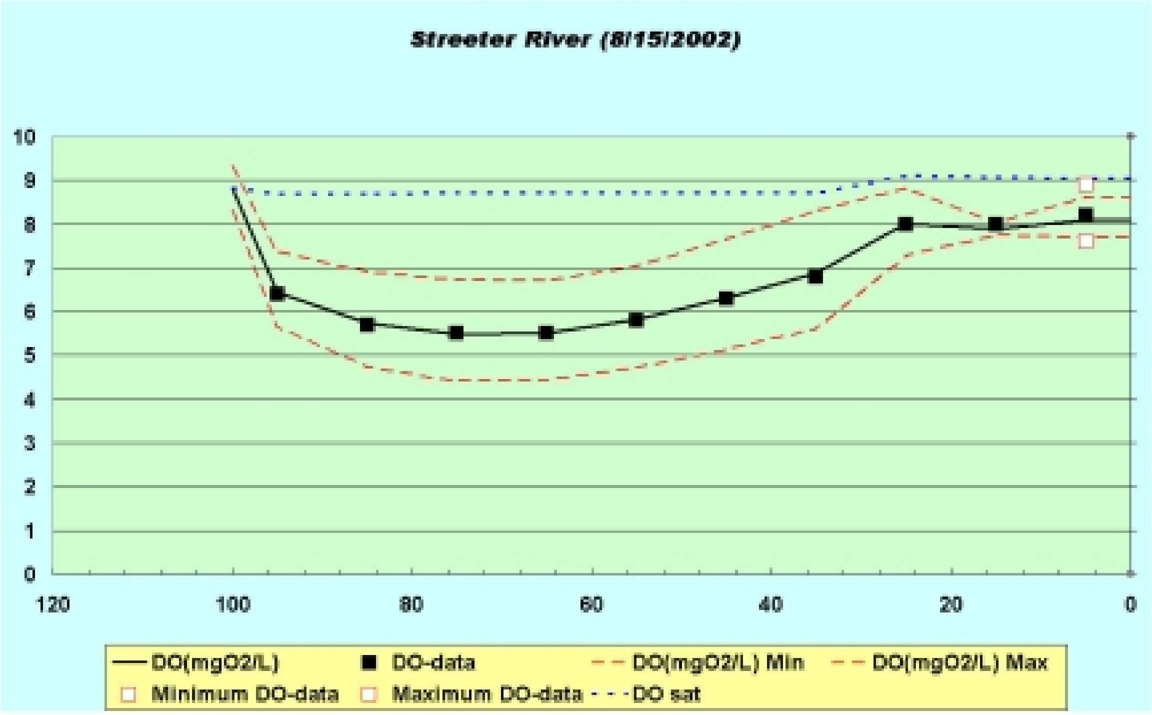


Figure 54. The plot of dissolved oxygen versus distance downstream in km.

The following series of variables are plotted:

Hydraulics Plots:

- **Travel Time**
- **Flow**
- **Velocity**
- **Depth**
- **Reaeration**

Temperature and state-variable plots:

- **Temperature**
- **Conductivity**
- **ISS (Inorganic suspended solids)**
- **Dissolved oxygen**
- **Detritus**
- **Slow CBOD**
- **Fast CBOD**
- **DON (Dissolved organic nitrogen)**
- **NH4 (Ammonia nitrogen)**
- **NO3 (Nitrate nitrogen)**
- **DOP (Dissolved organic phosphorus)**
- **Inorganic phosphorus**
- **Phytoplankton**
- **Bot Pl gD per m2 (Bottom algae in units of gD/m^2)**
- **Pathogen**
- **Alkalinity**
- **pH**

Additional State-variable plots:

- **Bot Pl mgA per m2 (Bottom algae in units of mgA/m^2)**
- **CBODu**
- **NH3**
- **TN and TP**
- **TSS**

Sediment-water plots:

- **SOD**
- **CH4 Sed Flux**
- **NH4 Sed Flux**
- **Inorg P Sed Flux**

6.2.17 Diel Charts

QUAL2K displays a series of charts that plot the model output and data versus time of day (in hours) for temperature and the model state variables.

Figure 55 shows an example of the diel plot for pH. The red line is the simulated pH (as displayed on the Diel Worksheet). The black squares are the measured data points that were entered on the Diel Data Worksheet. The plot is labeled with the river name, the date and the name of the reach that is plotted.

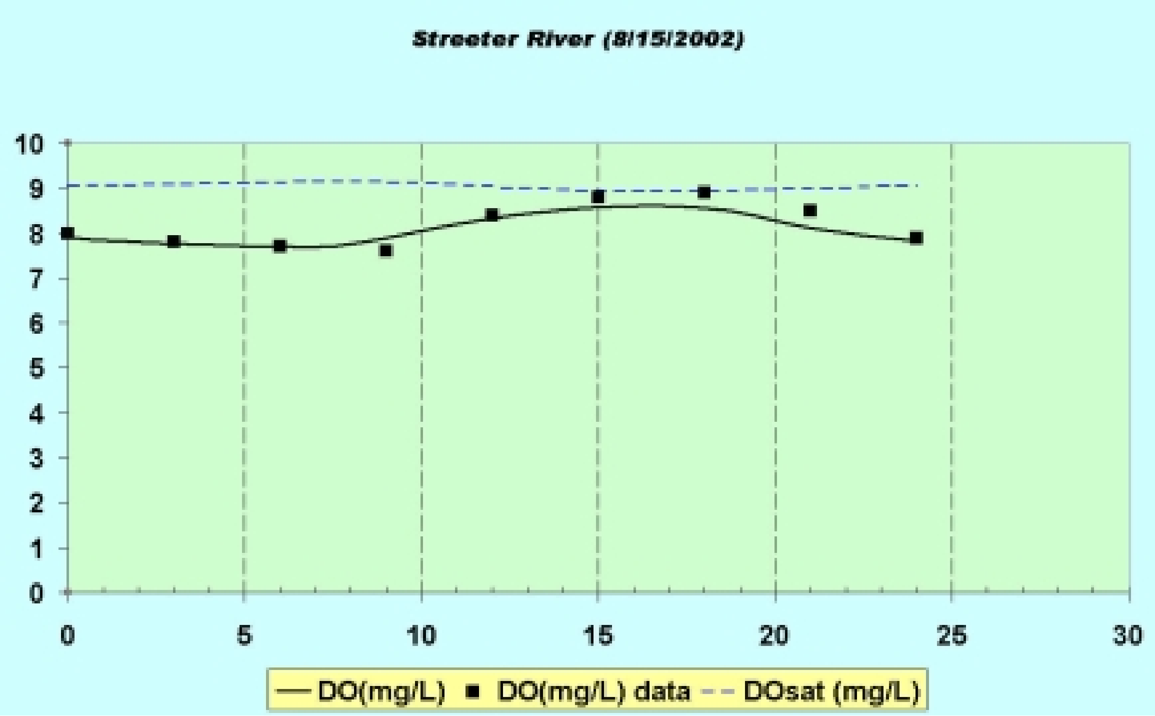


Figure 55. The diel plot of the Dissolved Oxygen versus time of day.

7 REFERENCES

Adams, E.E., D. J. Cosler, and K.R. Helfrich. 1987. "Analysis of evaporation data from heated ponds", cs-5171, research project 2385-1, Electric Power Research Institute, Palo Alto, California 94304. April.

Andrews and Rodvey, 1980. Heat exchange between water and tidal flats. D.G.M 24(2). (in German)

Barnwell, T.O., Brown, L.C., and Mareck, W. 1989). Application of Expert Systems Technology in Water Quality Modeling. *Water Sci. Tech.* 21(8-9):1045-1056.

Bejan, A. 1993. *Heat Transfer.* Wiley, New York, NY.

Bowie, G.L., Mills, W.B., Porcella, D.B., Campbell, C.L., Pagenkopf, J.R., Rupp, G.L., Johnson, K.M., Chan, P.W.H., Gherini, S.A. and Chamberlin, C.E. 1985. Rates, Constants, and Kinetic Formulations in Surface Water Quality Modeling. U.S. Envir. Prot. Agency, ORD, Athens, GA, ERL, EPA/600/3-85/040.

Brady, D.K., Graves, W.L., and Geyer, J.C. 1969. Surface Heat Exchange at Power Plant Cooling Lakes, Cooling Water Discharge Project Report, No. 5, Edison Electric Inst. Pub. No. 69-901, New York, NY.

Bras, R.L. 1990. Hydrology. Addison-Wesley, Reading, MA.

Brown, L.C., and Barnwell, T.O. 1987. The Enhanced Stream Water Quality Models QUAL2E and QUAL2E-UNCAS, EPA/600/3-87-007, U.S. Environmental Protection Agency, Athens, GA, 189 pp.

Brunt, D. 1932. Notes on radiation in the atmosphere: I. Quart J Royal Meteorol Soc 58:389-420.

Brutsaert, W. 1982. Evaporation into the atmosphere: theory, history, and applications. D.Reidel Publishing Co., Hingham MA, 299 p.

Butts, T. A. and Evans, R. L. 1983. Effects of Channel Dams on Dissolved Oxygen Concentrations in Northeastern Illinois Streams, Circular 132, State of Illinois, Dept. of Reg. and Educ., Illinois Water Survey, Urbana, IL.

Carslaw, H.S. and Jaeger, J.C. 1959. Conduction of Heat in Solids, Oxford Press, Oxford, UK, 510 pp.

Cengel, Y.A. 1998 *Heat Transfer: A Practical Approach.* New York, McGraw-Hill.

Chapra and Canale 2002. *Numerical Methods for Engineers, 4th Ed.* New York, McGraw-Hill.

Chapra, S.C. 1997. *Surface water quality modeling*. New York, McGraw-Hill.

Chow, V.T., Maidment, D.R., and Mays, L.W. 1988. *Applied Hydrology*. New York, McGraw-Hill, 592 pp.

Covar, A. P. 1976. "Selecting the Proper Reaeration Coefficient for Use in Water Quality Models." Presented at the U.S. EPA Conference on Environmental Simulation and Modeling, April 19-22, 1976, Cincinnati, OH.

Di Toro, D. M. and J. F. Fitzpatrick. 1993. Chesapeake Bay sediment flux model. Tech. Report EL-93-2, U.S. Army Corps of Engineers, Waterways Experiment Station, Vicksburg, Mississippi, 316 pp.

Di Toro, D.M, Paquin, P.R., Subburamu, K. and Gruber, D.A. 1991. Sediment Oxygen Demand Model: Methane and Ammonia Oxidation. J. Environ. Eng., 116(5):945-986.

Di Toro, D.M. 1978. Optics of Turbid Estuarine Waters: Approximations and Applications. Water Res. 12:1059-1068.

Di Toro, D.M. 2001. *Sediment Flux Modeling*. Wiley-Interscience, New York, NY.

Ecology. 2003. Shade.xls - a tool for estimating shade from riparian vegetation. Washington State Department of Ecology. http://www.ecy.wa.gov/programs/eap/models/

Edinger, J.E., Brady, D.K., and Geyer, J.C. 1974. Heat Exchange and Transport in the Environment. Report No. 14, EPRI Pub. No. EA-74-049-00-3, Electric Power Research Institute, Palo Alto, CA.

Finnemore, E.J. and Franzini, J.B. 2002. Fluid Mechanics with Engineering Applications, 10th Ed. New York, McGraw, Hill.

Geiger, R. 1965. The climate near the ground. Harvard University Press. Cambridge, MA.

Gordon, N.D, T.A. McMahon, and B.L. Finlayson. 1992. Stream Hydrology, An Introduction for Ecologists. Published by John Wiley and Sons.

Grigull, U. and Sandner, H. 1984. *Heat Conduction.* Springer-Verlag, New York, NY.

Harbeck, G. E., 1962, A practical field technique for measuring reservoir evaporation utilizing mass-transfer theory. US Geological Survey Professional Paper 272-E, 101-5.

Harned, H.S., and Hamer, W.J. 1933. “The Ionization Constant of Water.” *J. Am., Chem. Soc.* 51:2194.

Helfrich, K.R., E.E. Adams, A.L. Godbey, and D.R.F. Harleman. 1982. Evaluation of models for predicting evaporative water loss in cooling impoundments. Report CS-2325, Research project 1260-17. Electric Power Research Institute, Pala Alto, CA. March 1982.

Hutchinson, G.E. 1957. *A Treatise on Limnology, Vol. 1, Physics and Chemistry*. Wiley, New York, NY.

Jobson, H.E. 1977. Bed Conduction Computation for Thermal Models. J. Hydraul. Div. ASCE. 103(10):1213-1217.

Koberg, G.E. 1964. Methods to compute long-wave radiation from the atmosphere and reflected solar radiation from a water surface. US Geological Survey Professional Paper 272-F.

Kreith, F. and Bohn, M.S. 1986. *Principles of Heat Transfer, 4th Ed.* Harper and Row, New York, NY.

Laws, E. A. and Chalup, M. S. 1990. A Microalgal Growth Model. *Limnol. Oceanogr.* 35(3):597-608.

LI-COR, 2003. Radiation Measurement Instruments, LI-COR, Lincoln, NE, 30 pp.

Likens, G. E., and Johnson, N. M. (1969). "Measurements and analysis of the annual heat budget for sediments of two Wisconsin lakes." *Limnol. Oceanogr.*, 14(1):115-135.

Mackay, D. and Yeun, A.T.K. 1983. Mass Transfer Coefficient Correlations for Volatilization of Organic Solutes from Water. *Environ. Sci. Technol.* 17:211-233.

Marciano, J.K. and G.E. Harbeck. 1952. Mass transfer studies in water loss investigation: Lake Hefner studies. Geological Circular 229. U.S. Geological Survey, Washington DC.

Meeus, J. 1999. Astronomical algorithms. Second edition. Willmann-Bell, Inc. Richmond, VA.

Mills, A.F. 1992. *Heat Transfer.* Irwin, Homewood, IL.

Nakshabandi, G.A. and H. Kohnke. 1965. Thermal conductivity and diffusivity of soils as related to moisture tension and other physical properties. Agr. Met. Vol 2.

Plummer, L.N. and Busenberg, E. 1982. The Solubilities of Calcite, Aragonite and Vaterite in CO2-H2O Solutions Between 0 and 90 oC, and an Evaluation of the Aqueous Model for the System CaCO3–CO2–H2O. *Geochim. Cosmochim.* 46:1011-1040.

Raudkivi, A. I. 1979. *Hydrology*. Pergamon, Oxford, England.

Redfield, A.C., Ketchum, B.H. and Richards, F.A. 1963. The Influence of Organisms on the Composition of Seawater, in *The Sea*, M.N. Hill, ed. Vol. 2, pp. 27-46, Wiley-Interscience, NY.

Riley, G.A. 1956. Oceanography of Long Island Sound 1952-1954. II. Physical Oceanography, Bull. Bingham Oceanog. Collection 15, pp. 15-16.

Rosgen, D. 1996. Applied river morphology. Wildland Hydrology publishers. Pagosa Springs, CO.

Rutherford, J.C., Scarsbrook, M.R. and Broekhuizen, N. 2000. Grazer Control of Stream Algae: Modeling Temperature and Flood Effects. *J. Environ. Eng.* 126(4):331-339.

Ryan, P.J. and D.R.F. Harleman. 1971. Prediction of the annual cycle of temperature changes in a stratified lake or reservoir. Mathematical model and user’s manual. Ralph M. Parsons Laboratory Report No. 137. Massachusetts Institute of Technology. Cambridge, MA.

Ryan, P.J. and K.D. Stolzenbach.1972. Engineering aspects of heat disposal from power generation, (D.R.F. Harleman, ed.). R.M. Parson Laboratory for Water Resources and Hydrodynamics, Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA

Shanahan, P. 1984. Water temperature modeling: a practical guide. In: Proceedings of stormwater and water quality model users group meeting, April 12-13, 1984. USEPA, EPA-600/9-85-003. (users.rcn.com/shanahan.ma.ultranet/TempModeling.pdf)

Stumm, W. and Morgan, J.J. 1996. Aquatic Chemistry, New York, Wiley-Interscience, 1022 pp.

Thomann, R.V. and Mueller, J.A. 1987. Principles of Surface Water Quality Modeling and Control. New York, Harper-Collins.

TVA, 1972. Heat and mass transfer between a water surface and the atmosphere. Water Resources Research, Laboratory Report No. 14. Engineering Laboratory, Division of Water Control Planning, Tennessee Valley Authority, Norris TN.

Wood, K.G. 1974. Carbon Dioxide Diffusivity Across the Air-Water Interface. *Archiv für Hydrobiol.* 73(1):57-69.

8 APPENDIX A: NOMENCLATURE

Symbol	Definition	Units
$A_{acres,i}$	surface area of reach i	acres
$[CO_2]_s$	saturation concentration of carbon dioxide	mole/L
$[CO_3^{2-}]$	carbonate ion	mole/L
$[H^+]$	hydronium ion	mole/L
$[H_2CO_3*]$	sum of dissolved carbon dioxide and carbonic acid	mole/L
$[HCO_3^-]$	bicarbonate ion	mole/L
$[OH^-]$	hydroxyl ion	mole/L
a	velocity rating curve coefficient	dimensionless
a''	mean atmospheric transmission coefficient after scattering and absorption	dimensionless
a'	mean atmospheric transmission coefficient	dimensionless
a_1	atmospheric molecular scattering coefficient for radiation transmission	dimensionless
A_a	atmospheric long-wave radiation coefficient	dimensionless
A_b	atmospheric long-wave radiation coefficient	$mmHg^{-0.5}$ or $mb^{-0.5}$
a_b	bottom algae	gD/m^2
A_c	cross-sectional area	m^2
a_d	coefficient correcting dam reaeration for water quality	dimensionless
Alk	alkalinity	eq L^{-1} or $mgCaCO_3/L$
a_p	phytoplankton	mgC/L
a_p	phytoplankton concentration	mgA/m^3
a_t	atmospheric attenuation	dimensionless
a_{tc}	atmospheric transmission coefficient	dimensionless
B	average reach width	m
b	velocity rating curve exponent	dimensionless
B_0	bottom width	m
b_d	coefficient correcting dam reaeration for dam type	dimensionless
c_1	Bowen's coefficient	0.47 mmHg/^{o}C
c_f	fast reacting CBOD	mgO_2/L
c_f	fast CBOD in the overlying water	gO_2/m^3
$C_{gb}(T)$	temperature-dependent maximum photosynthesis rate	$gD/(m^2\ d)$
$CH_{4,1}$	methane concentration in the aerobic sediment layer	gO_2/m^3
C_L	fraction of sky covered with clouds	dimensionless
$c_{nps,i,j}$	*j*th non-point source concentration for reach *i*	^{o}C
C_p	specific heat of water	cal/(g ^{o}C)
$c_{ps,i,j}$	*j*th point source concentration for reach *i*	^{o}C
c_s	slowly reacting CBOD	mgO_2/L
C_s	saturation concentration of methane	mgO_2/L
$CSOD$	the amount of oxygen demand generated by methane oxidation	$gO_2/m^2/d$
c_T	total inorganic carbon	mole L^{-1}
$c'_{T,i-1}$	concentration of inorganic carbon entering reach below a dam	mgO_2/L

d	dust attenuation coefficient	dimensionless
D_d	pore water diffusion coefficient	m^2/d
D_p	diffusion coefficient for bioturbation	m^2/d
E'_i	bulk dispersion coefficient between reaches *i* and *i* + 1	m^3/d
e_{air}	air vapor pressure	mm Hg
$elev$	elevation above sea level	m
E_n	numerical dispersion	m^2/d
$E_{p,i}$	longitudinal dispersion between reaches *i* and *i* + 1	m^2/s
$eqtime$	equation of time: the difference between mean solar time and true solar time when located on the reference longitude of the time zone considered	minutes
e_s	saturation vapor pressure at water surface	mmHg
f	photoperiod	fraction of day
f_{dai}	fraction of ammonium in dissolved form in sediment layer *i*	dimensionless
f_{dpi}	fraction of inorganic phosphorus in dissolved form in sediment layer *i*	dimensionless
F_{Lp}	phytoplankton growth attenuation due to light	dimensionless
F_{oxcf}	attenuation of CBOD oxidation due to low oxygen	dimensionless
F_{oxdn}	enhancement of denitrification at low oxygen concentration	dimensionless
F_{oxna}	attenuation due to low oxygen	dimensionless
F_{oxrb}	attenuation due to low oxygen	dimensionless
F_{oxrp}	attenuation due to low oxygen	dimensionless
f_{pai}	fraction of ammonium in particulate form in sediment layer *i*	dimensionless
f_{ppi}	fraction of inorganic phosphorus in particulate form in sediment layer *i*	dimensionless
F_u	fraction unionized ammonia	
g	acceleration due to gravity	= 9.81 m/s^2
gX	mass of element X	g
H	water depth	m
H	water depth	m
H_d	drop in water elevation for a dam	m
H_i	thickness of sediment layer *i*	m
H_{sed}	sediment thickness	cm
$I(0)$	solar radiation at water surface	cal/cm^2/d
I_0	extraterrestrial radiation	cal/cm^2/d
J_{an}	net atmospheric longwave radiation flux	cal/(cm^2 d)
J_{br}	longwave back radiation flux from water	cal/(cm^2 d)
J_c	conduction flux	cal/(cm^2 d)
$J_{C,G1}$	flux of labile dissolved carbon	gO$_2$/m^2/d
J_{CH4}	Flux of methane from sediments to the overlying water	gO$_2$/m^2/d
$J_{CH4,d}$	flux of dissolved methane generated in anaerobic sediments corrected for methane gas formation	gO$_2$/m^2/d
$J_{CH4,T}$	carbon diagenesis flux corrected for denitrification	gO$_2$/m^2/d
J_e	evaporation flux	cal/(cm^2 d)
J_h	air-water heat flux	cal/(cm^2 d)

J_N	nitrogen diagenesis flux	gN/m^2/d
$J_{O2,dn}$	oxygen equivalents consumed during denitrification	gO$_2$/m^2/d
J_P	phosphorus diagenesis flux	gP/m^2/d
$J_{POC,G1}$	the flux of labile POC delivered to the anaerobic sediment layer	gO$_2$/m^2/d
J_{POM}	the downward flux of POM	gD m^{-2} d^{-1}
J_{si}	sediment-water heat flux	cal/(cm^2 d)
J_{sn}	net solar shortwave radiation flux	cal/(cm^2 d)
$k(T)$	temperature dependent first-order reaction rate	/d
K_1	acidity constant for dissociation of carbonic acid	
K_2	acidity constant for dissociation of bicarbonate	
K_a	equilibrium coefficient for ammonium dissociation	
$k_a(T)$	temperature-dependent oxygen reaeration coefficient	/d
$k_{ac}(T)$	temperature-dependent carbon dioxide reaeration coefficient	/d
$k_{db}(T)$	temperature-dependent bottom algae death rate	/d
$k_{dc}(T)$	temperature-dependent fast CBOD oxidation rate	/d
$k_{dn}(T)$	temperature-dependent denitrification rate	/d
$k_{dp}(T)$	temperature-dependent phytoplankton death rate	/d
$k_{dt}(T)$	temperature-dependent detritus dissolution rate	/d
$k_{dx}(T)$	temperature-dependent pathogen die-off rate	/d
k_e	light extinction coefficient	/m^1
k_{eb}	a background coefficient accounting for extinction due to water and color	/m
$k_{gp}(T)$	maximum photosynthesis rate at temperature T	/d
K_H	Henry's constant	mole/(L atm)
$k_{hc}(T)$	temperature-dependent slow CBOD hydrolysis rate	/d
$k_{hn}(T)$	temperature-dependent organic nitrogen hydrolysis rate	/d
k_{hnxb}	preference coefficient of bottom algae for ammonium	mgN/m^3
k_{hnxp}	preference coefficient of phytoplankton for ammonium	mgN/m^3
$k_{hp}(T)$	temperature-dependent organic phosphorus hydrolysis rate	/d
K_{L12}	pore water diffusion mass transfer coefficient	m/d
K_{Lb}	bottom algae light parameter	
K_{Lp}	phytoplankton light parameter	ly/d
$K_{M,Dp}$	oxygen half-saturation constant for bioturbation	gO$_2$/m^3
$k_{na}(T)$	temperature-dependent nitrification rate for ammonia nitrogen	/d
K_{NH4}	ammonium half-saturation constant	gN/m^3
$K_{NH4,O2}$	oxygen half-saturation constant	mgO$_2$/L
$k_{POC,G1}$	the mineralization rate of labile POC	d^{-1}
$k_{rb}(T)$	temperature-dependent bottom algae respiration rate	/d
$k_{rp}(T)$	temperature-dependent phytoplankton respiration rate	/d
k_{sNb}	nitrogen half-saturation constant for bottom algae	μgN/L
k_{sNp}	nitrogen half-saturation constant for phytoplankton	μgN/L
K_{socf}	parameter for oxygen dependency of fast CBOD oxidation	
K_{sodn}	parameter for oxygen dependency of denitrification	

K_{sona}	parameter for oxygen dependency of nitrification	
k_{sPb}	phosphorus half-saturation constant for bottom algae	µgP/L
k_{sPp}	phosphorus half-saturation constant for phytoplankton	µgP/L
K_w	acidity constant for dissociation of water	
L_{at}	latitude	radians
L_{lm}	longitude of local meridian	degrees
localTime	local standard time	minutes
L_{sm}	longitude of standard meridian	degrees
m	optical air mass	dimensionless
mgY	mass of element Y	mg
m_i	inorganic suspended solids	mgD/L
m_i	the solids concentration in sediment layer *i*	gD/m^3
m_o	detritus	mgD/L
m_o	detritus concentration	gD/m^3
n	Manning roughness coefficient	
n_a	ammonia nitrogen	mgN/L
n_a	the ammonium concentration in the overlying water	mgN/m^3
n_{au}	unionized ammonia nitrogen	mgN/L
n_{fac}	atmospheric turbidity factor	dimensionless
$NH_{4,i}$	the concentration of total ammonium in sediment layer *i*	gN/m^3
n_n	nitrate nitrogen	mgN/L
n_n	nitrate concentration in the overlying water	mgN/m^3
n_o	dissolved organic nitrogen	mgN/L
$NO_{3,i}$	nitrate concentration in layer *i*	gN/m^3
npai	total number of non-point abstractions outflows from reach *i*	dimensionless
npsi	total number of non-point sources inflows to reach *i*	dimensionless
NSOD	the amount of oxygen demand generated by nitrification	gO$_2$/m^2/d
o	dissolved oxygen	mgO$_2$/L
o	the dissolved oxygen concentration in the overlying water	gO$_2$/m^3
o'_{i-1}	oxygen concentration entering a reach below a dam	mgO$_2$/L
o_{crit}	critical oxygen concentration for sediment phosphorus sorption	gO$_2$/m^3
$o_s(T, \mathrm{elev})$	saturation concentration of oxygen at temperature, *T*, and elevation above sea level, *elev*	mgO$_2$/L
P	wetted perimeter	m
P_{ab}	preference for ammonium as a nitrogen source for bottom algae	dimensionless
pai	total number of point abstractions from reach *i*	dimensionless
P_{ap}	preference for ammonium as a nitrogen source for phytoplankton	dimensionless
$PAR(z)$	photosynthetically available radiation (PAR) at depth *z* below water surface	ly/d
p_{atm}	atmospheric pressure	mm Hg
$\mathrm{p}_{\mathrm{CO}_2}$	atmospheric partial pressure of carbon dioxide	atm
p_i	inorganic phosphorus	µgP/L
p_i	the inorganic phosphorus in the overlying water	mgP/m^3
p_o	dissolved organic phosphorus	µgP/L

$PO_{4,i}$	the concentration of total inorganic phosphorus in sediment layer *i*	gP/m^3
$POC_{2,G1}$	the concentration of the labile fraction of POC in the anaerobic sediment layer	gO_2/m^3
POC_R	reference G1 concentration for bioturbation	gC/m^3
psi	total number of point sources to reach *i*	dimensionless
p_{wc}	mean daily atmospheric precipitable water content	
Q	flow	m^3/s or m^3/d
$Q_{ab,i}$	total outflow from reach due to point and nonpoint abstractions	m^3/d
Q_i	outflow from reach *i* into reach *i* + 1	m^3/d
$Q_{in,i}$	total inflow into reach from point and nonpoint sources	m^3/d
$Q_{npa,i,j}$	*j*th non-point abstraction outflow from reach *i*	m^3/d
$Q_{nps,i,j}$	*j*th non-point source inflow to reach *i*	m^3/d
$Q_{pa,i,j}$	*j*th point abstraction outflow from reach *i*	m^3/d
$Q_{ps,i,j}$	*j*th point source inflow to reach *i*	m^3/d
r	normalized radius of earth’s orbit (i.e., ratio of actual earth-sun distance to mean earth-sun distance	dimensionless
r_{cndn}	ratio of oxygen equivalents lost per nitrate nitrogen that is denitrified	gO_2/gN
r_d	ratio of deficit above and below dam	dimensionless
r_{da}	the ratio of dry weight to chlorophyll *a*	gD/mgA
R_L	longwave reflection coefficient	dimensionless
r_{oc}	ratio of oxygen consumed per organic carbon oxidized to carbon dioxide	gO_2/gC
r_{oca}	ratio of oxygen generated per organic carbon produced by photosynthesis when nitrate is taken up	gO_2/gC
r_{ocn}	ratio of oxygen generated per organic carbon produced by photosynthesis when ammonia is taken up	gO_2/gC
r_{on}	the ratio of oxygen to nitrogen consumed during nitrification	$= 4.57\ gO_2/gN$
r_{on}'	ratio of oxygen consumed per ammonia nitrogen converted by nitrification/denitrification to nitrogen gas	gO_2/gN
r_{on}’	the ratio of oxygen to nitrogen consumed for total conversion of ammonium to nitrogen gas via nitrification/denitrification	$= 1.714\ gO_2/gN$
R_s	albedo or reflectivity (fraction of solar radiation that is reflected)	dimensionless
S	channel slope	dimensionless
s	chloride	mgCl/L
s	mass transfer coefficient between the water and the aerobic sediments	m/d
S_0	bottom slope	m/m
$S_{b,i}$	sources and sinks of constituent due to reactions for bottom algae	$gD/m^2/d$
S_i	sources and sinks of constituent due to reactions and mass transfer mechanisms for water constituents	$g/m^3/d$ or $mg/m^3/d$
SOD	the sediment oxygen demand	$gO_2/m^2/d$
SOD_s	the supplemental sediment oxygen demand	$gO_2/m^2/d$

SOD_t	total sediment oxygen demand = $SOD + SOD_s$	$gO_2/m^2/d$
s_s	channel side slope	m/m
t	time	d
T	water temperature	^oC
T	water temperature	^oC
$T_{,w,f}$	water temperature	^oF
T_a	absolute temperature	K
T_{air}	air temperature	^oC
$T_{air,f}$	air temperature	^oF
T_d	dew-point temperature	^oC
T_i	temperature in reach i	^oC
timezone	time zone indicates local time zone in relation to Greenwich Mean Time (GMT) (negative in western hemisphere)	hours
$T_{nps,i,j}$	jth non-point source temperature for reach i	^oC
$T_{ps,i,j}$	jth point source temperature for reach i	^oC
trueSolarTime	time determined from actual position of the sun in the sky	minutes
$T_{s,i}$	temperature of bottom sediment	^oC
t_{sr}	time of sunrise	Hr
t_{ss}	time of sunset	Hr
T_{std}	standard time	Hr
$t_{t,i}$	travel time from headwater to end of reach i	D
U	mean velocity	m/s
U_i^*	shear velocity	m/s
U_w	wind speed	m/s
$U_{w,mph}$	wind speed	mph
$U_{w,z}$	wind speed at height z_w above water surface	m/s
v_a	phytoplankton settling velocity	m/d
v_a	phytoplankton settling velocity	m/d
v_{dt}	detritus settling velocity	m/d
v_{dt}	detritus settling velocity	m/d
v_i	inorganic suspended solids settling velocity	m/d
V_i	volume of i^{th} reach	m^3
v_p	pathogen settling velocity	m/d
v_{pc}	non-living particulate organic carbon settling velocity	m/d
W_0	solar constant	2851 $cal/cm^2/d$
w_2	the burial velocity	m/d
$W_{h,i}$	net heat load from point and non-point sources into reach i	cal/d
W_i	external loading of constituent to reach i	g/d or mg/d
x	pathogen	cfu/100 mL
z_w	height above water for wind speed measurements	m

Greek:

Symbol	Definition	Units
α	depth rating curve coefficient	dimensionless
α	sun's altitude	radians
α_d	sun's altitude	degrees
α_s	sediment thermal diffusivity	cm^2/s
α_0	fraction of total inorganic carbon in carbon dioxide	dimensionless
α_1	fraction of total inorganic carbon in bicarbonate	dimensionless
α_2	fraction of total inorganic carbon in carbonate	dimensionless
α_i	effect of inorganic suspended solids on light attenuation	L/mgD/m
α_o	effect of particulate organic matter on light attenuation	L/mgD/m
α_p	linear effect of chlorophyll on light attenuation	L/µgA/m
α_{pn}	non-linear effect of chlorophyll on light attenuation	$(L/\mu gA)^{2/3}/m$
β	depth rating curve exponent	dimensionless
δ	solar declination	radians
$\Delta\pi_{PO4,1}$	factor that increases the aerobic sediment layer partition coefficient relative to the anaerobic coefficient	dimensionless
$\Delta\theta_v$	virtual temperature difference between the water and air	°F
Δt_s	difference between standard and local civic time	hr
Δx_i	length of *i*th reach	m
ε	emissivity of water	dimensionless
ε_{clear}	emissivity of longwave radiation from the sky with no clouds	0-1
ε_{sky}	emissivity of longwave radiation from the sky with clouds	0-1
ε_a	estimated error	%
ϕ_{Lb}	bottom algae light attenuation	0-1
ϕ_{Lp}	phytoplankton light attenuation	0-1
ϕ_{Nb}	bottom algae nutrient attenuation factor	0-1
ϕ_{Np}	phytoplankton nutrient attenuation factor	0-1
μ_p	phytoplankton photosynthesis rate	/d
ρ	density of water	g/cm^3
σ	Stefan-Boltzmann constant	$11.7\text{x}10^{-8}$ cal/(cm^2 d K^4)
τ	local hour angle of sun	radians
τ_i	residence time of i^{th} reach	d
θ	temperature coefficient for zero and first-order reactions	dimensionless
θ_{am}	elevation adjusted optical air mass	
ω_{12}	the bioturbation mass transfer coefficient between the sediment layers	m/d
π_{ai}	the partition coefficient for ammonium in sediment layer *i*	m^3/gD
π_{pi}	the partition coefficient for inorganic phosphorus in sediment layer *i*	m^3/gD
θ_{CH4}	temperature correction factor for sediment methane oxidation	dimensionless

θ_{Dd}	temperature coefficient for porewater diffusion	dimensionless
θ_{Dp}	temperature coefficient for bioturbation diffusion	dimensionless
θ_{NH4}	temperature correction factor for sediment nitrification	dimensionless
θ_{NO3}	sediment denitrification temperature correction factor	dimensionless
$\theta_{POC,G1}$	temperature correction factor for labile POC mineralization	dimensionless
$\kappa_{CH4,1}$	the reaction velocity for methane oxidation in the aerobic sediments	m/d
$\kappa_{NH4,1}$	the reaction velocity for nitrification in the aerobic sediments	m/d
$\kappa_{NO3,i}$	denitrification reaction velocity sediment layer i	m/d

9 APPENDIX B: SOLAR POSITION, SUNRISE, AND SUNSET CALCULATIONS

The sunrise/sunset and solar position functions are a VBA translation of NOAA's sunrise/sunset calculator and NOAA's solar position calculator at the following web pages:

- http://www.srrb.noaa.gov/highlights/sunrise/sunrise.html
- http://www.srrb.noaa.gov/highlights/sunrise/azel.html

The calculations in the NOAA Sunrise/Sunset and Solar Position Calculators are based on equations from Astronomical Algorithms, by Jean Meeus.The sunrise and sunset results have been verified by NOAA to be accurate to within a minute for locations between +/- 72° latitude, and within 10 minutes outside of those latitudes.

Five main functions are included for use from Excel worksheets or VBA programs:

- sunrise(lat, lon, year, month, day, timezone, dlstime) calculates the local time of sunrise for a location and date
- solarnoon(lat, lon, year, month, day, timezone, dlstime) calculates the local time of solar noon for a location and date (the time when the sun crosses the meridian)
- sunset(lat, lon, year, month, day, timezone, dlstime) calculates the local time of sunset for a location and date
- solarazimuth(lat, lon, year, month, day, hour, minute, second, timezone, dlstime) calculates the solar azimuth for a location, date, and time (degrees clockwise from north to the point on the horizon directly below the sun)
- solarelevation(lat, lon, year, month, day, hour, minute, second, timezone, dlstime) calculates the solar elevation for a location, date, and time (degrees vertically from horizon to the sun)

The sign convention for the main functions and subroutine is:

- positive latitude decimal degrees for northern hemisphere
- negative longitude degrees for western hemisphere
- negative time zone hours for western hemisphere

The Excel/VBA functions and subroutines for solar position, sunrise, and sunset times are as follows:

```
Option Explicit

Function radToDeg(angleRad)
'// Convert radian angle to degrees

        radToDeg = (180# * angleRad / Application.WorksheetFunction.Pi())

End Function

Function degToRad(angleDeg)
'// Convert degree angle to radians

        degToRad = (Application.WorksheetFunction.Pi() * angleDeg / 180#)

End Function

Function calcJD(year, month, day)

'***********************************************************************/
'* Name:    calcJD
'* Type:    Function
'* Purpose: Julian day from calendar day
'* Arguments:
'*   year : 4 digit year
```

```
'*   month: January = 1
'*   day  : 1 - 31
'* Return value:
'*   The Julian day corresponding to the date
'* Note:
'*   Number is returned for start of day.  Fractional days should be
'*   added later.
'***********************************************************************/

Dim A As Double, B As Double, jd As Double

        If (month <= 2) Then
         year = year - 1
         month = month + 12
        End If

        A = Application.WorksheetFunction.Floor(year / 100, 1)
        B = 2 - A + Application.WorksheetFunction.Floor(A / 4, 1)

        jd = Application.WorksheetFunction.Floor(365.25 * (year + 4716), 1) + _
             Application.WorksheetFunction.Floor(30.6001 * (month + 1), 1) + day + B - 1524.5
        calcJD = jd

'gp put the year and month back where they belong
        If month = 13 Then
         month = 1
         year = year + 1
        End If
        If month = 14 Then
         month = 2
         year = year + 1
        End If

End Function

Function calcTimeJulianCent(jd)

'***********************************************************************/
'* Name:    calcTimeJulianCent
'* Type:    Function
'* Purpose: convert Julian Day to centuries since J2000.0.
'* Arguments:
'*   jd : the Julian Day to convert
'* Return value:
'*   the T value corresponding to the Julian Day
'***********************************************************************/

Dim t As Double

        t = (jd - 2451545#) / 36525#
        calcTimeJulianCent = t

End Function

Function calcJDFromJulianCent(t)

'***********************************************************************/
'* Name:    calcJDFromJulianCent
'* Type:    Function
'* Purpose: convert centuries since J2000.0 to Julian Day.
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   the Julian Day corresponding to the t value
'***********************************************************************/

Dim jd As Double

        jd = t * 36525# + 2451545#
        calcJDFromJulianCent = jd

End Function

Function calcGeomMeanLongSun(t)

'***********************************************************************/
'* Name:    calGeomMeanLongSun
'* Type:    Function
'* Purpose: calculate the Geometric Mean Longitude of the Sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
```

```
'* Return value:
'*   the Geometric Mean Longitude of the Sun in degrees
'*********************************************************************/

Dim l0 As Double

        l0 = 280.46646 + t * (36000.76983 + 0.0003032 * t)
        Do
           If (l0 <= 360) And (l0 >= 0) Then Exit Do
           If l0 > 360 Then l0 = l0 - 360
           If l0 < 0 Then l0 = l0 + 360
        Loop

        calcGeomMeanLongSun = l0

End Function

Function calcGeomMeanAnomalySun(t)

'*********************************************************************/
'* Name:    calGeomAnomalySun
'* Type:    Function
'* Purpose: calculate the Geometric Mean Anomaly of the Sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   the Geometric Mean Anomaly of the Sun in degrees
'*********************************************************************/

Dim m As Double

        m = 357.52911 + t * (35999.05029 - 0.0001537 * t)
        calcGeomMeanAnomalySun = m

End Function

Function calcEccentricityEarthOrbit(t)

'*********************************************************************/
'* Name:    calcEccentricityEarthOrbit
'* Type:    Function
'* Purpose: calculate the eccentricity of earth's orbit
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   the unitless eccentricity
'*********************************************************************/

Dim e As Double

        e = 0.016708634 - t * (0.000042037 + 0.0000001267 * t)
        calcEccentricityEarthOrbit = e

End Function

Function calcSunEqOfCenter(t)

'*********************************************************************/
'* Name:    calcSunEqOfCenter
'* Type:    Function
'* Purpose: calculate the equation of center for the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   in degrees
'*********************************************************************/

Dim m As Double, mrad As Double, sinm As Double, sin2m As Double, sin3m As Double
Dim c As Double

        m = calcGeomMeanAnomalySun(t)

        mrad = degToRad(m)
        sinm = Sin(mrad)
        sin2m = Sin(mrad + mrad)
        sin3m = Sin(mrad + mrad + mrad)

        c = sinm * (1.914602 - t * (0.004817 + 0.000014 * t)) _
            + sin2m * (0.019993 - 0.000101 * t) + sin3m * 0.000289

        calcSunEqOfCenter = c
```

```
End Function

Function calcSunTrueLong(t)

'***********************************************************************/
'* Name:    calcSunTrueLong
'* Type:    Function
'* Purpose: calculate the true longitude of the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun's true longitude in degrees
'***********************************************************************/

Dim l0 As Double, c As Double, O As Double

        l0 = calcGeomMeanLongSun(t)
        c = calcSunEqOfCenter(t)

        O = l0 + c
        calcSunTrueLong = O

End Function

Function calcSunTrueAnomaly(t)

'***********************************************************************/
'* Name:    calcSunTrueAnomaly (not used by sunrise, solarnoon, sunset)
'* Type:    Function
'* Purpose: calculate the true anamoly of the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun's true anamoly in degrees
'***********************************************************************/

Dim m As Double, c As Double, v As Double

        m = calcGeomMeanAnomalySun(t)
        c = calcSunEqOfCenter(t)

        v = m + c
        calcSunTrueAnomaly = v

End Function

Function calcSunRadVector(t)

'***********************************************************************/
'* Name:    calcSunRadVector (not used by sunrise, solarnoon, sunset)
'* Type:    Function
'* Purpose: calculate the distance to the sun in AU
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun radius vector in AUs
'***********************************************************************/

Dim v As Double, e As Double, r As Double

        v = calcSunTrueAnomaly(t)
        e = calcEccentricityEarthOrbit(t)

        r = (1.000001018 * (1 - e * e)) / (1 + e * Cos(degToRad(v)))
        calcSunRadVector = r

End Function

Function calcSunApparentLong(t)

'***********************************************************************/
'* Name:    calcSunApparentLong (not used by sunrise, solarnoon, sunset)
'* Type:    Function
'* Purpose: calculate the apparent longitude of the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun's apparent longitude in degrees
'***********************************************************************/
```

```
Dim O As Double, omega As Double, lambda As Double

        O = calcSunTrueLong(t)

        omega = 125.04 - 1934.136 * t
        lambda = O - 0.00569 - 0.00478 * Sin(degToRad(omega))
        calcSunApparentLong = lambda

End Function

Function calcMeanObliquityOfEcliptic(t)

'***********************************************************************/
'* Name:    calcMeanObliquityOfEcliptic
'* Type:    Function
'* Purpose: calculate the mean obliquity of the ecliptic
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   mean obliquity in degrees
'***********************************************************************/

Dim seconds As Double, e0 As Double

        seconds = 21.448 - t * (46.815 + t * (0.00059 - t * (0.001813)))
        e0 = 23# + (26# + (seconds / 60#)) / 60#
        calcMeanObliquityOfEcliptic = e0

End Function

Function calcObliquityCorrection(t)

'***********************************************************************/
'* Name:    calcObliquityCorrection
'* Type:    Function
'* Purpose: calculate the corrected obliquity of the ecliptic
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   corrected obliquity in degrees
'***********************************************************************/

Dim e0 As Double, omega As Double, e As Double

        e0 = calcMeanObliquityOfEcliptic(t)

        omega = 125.04 - 1934.136 * t
        e = e0 + 0.00256 * Cos(degToRad(omega))
        calcObliquityCorrection = e

End Function

Function calcSunRtAscension(t)

'***********************************************************************/
'* Name:    calcSunRtAscension (not used by sunrise, solarnoon, sunset)
'* Type:    Function
'* Purpose: calculate the right ascension of the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun's right ascension in degrees
'***********************************************************************/

Dim e As Double, lambda As Double, tananum As Double, tanadenom As Double
Dim alpha As Double

        e = calcObliquityCorrection(t)
        lambda = calcSunApparentLong(t)

        tananum = (Cos(degToRad(e)) * Sin(degToRad(lambda)))
        tanadenom = (Cos(degToRad(lambda)))

'original NOAA code using javascript Math.Atan2(y,x) convention:
'        var alpha = radToDeg(Math.atan2(tananum, tanadenom));
'        alpha = radToDeg(Application.WorksheetFunction.Atan2(tananum, tanadenom))

'translated using Excel VBA Application.WorksheetFunction.Atan2(x,y) convention:
        alpha = radToDeg(Application.WorksheetFunction.Atan2(tanadenom, tananum))

        calcSunRtAscension = alpha
```

```
End Function

Function calcSunDeclination(t)

'***********************************************************************/
'* Name:    calcSunDeclination
'* Type:    Function
'* Purpose: calculate the declination of the sun
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   sun's declination in degrees
'***********************************************************************/

Dim e As Double, lambda As Double, sint As Double, theta As Double

        e = calcObliquityCorrection(t)
        lambda = calcSunApparentLong(t)

        sint = Sin(degToRad(e)) * Sin(degToRad(lambda))
        theta = radToDeg(Application.WorksheetFunction.Asin(sint))
        calcSunDeclination = theta

End Function

Function calcEquationOfTime(t)

'***********************************************************************/
'* Name:    calcEquationOfTime
'* Type:    Function
'* Purpose: calculate the difference between true solar time and mean
'*     solar time
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'* Return value:
'*   equation of time in minutes of time
'***********************************************************************/

Dim epsilon As Double, l0 As Double, e As Double, m As Double
Dim y As Double, sin2l0 As Double, sinm As Double
Dim cos2l0 As Double, sin4l0 As Double, sin2m As Double, Etime As Double

        epsilon = calcObliquityCorrection(t)
        l0 = calcGeomMeanLongSun(t)
        e = calcEccentricityEarthOrbit(t)
        m = calcGeomMeanAnomalySun(t)

        y = Tan(degToRad(epsilon) / 2#)
        y = y ^ 2

        sin2l0 = Sin(2# * degToRad(l0))
        sinm = Sin(degToRad(m))
        cos2l0 = Cos(2# * degToRad(l0))
        sin4l0 = Sin(4# * degToRad(l0))
        sin2m = Sin(2# * degToRad(m))

        Etime = y * sin2l0 - 2# * e * sinm + 4# * e * y * sinm * cos2l0 _
                - 0.5 * y * y * sin4l0 - 1.25 * e * e * sin2m

        calcEquationOfTime = radToDeg(Etime) * 4#

End Function

Function calcHourAngleSunrise(lat, SolarDec)

'***********************************************************************/
'* Name:    calcHourAngleSunrise
'* Type:    Function
'* Purpose: calculate the hour angle of the sun at sunrise for the
'*         latitude
'* Arguments:
'*   lat : latitude of observer in degrees
'* solarDec : declination angle of sun in degrees
'* Return value:
'*   hour angle of sunrise in radians
'***********************************************************************/

Dim latrad As Double, sdRad As Double, HAarg As Double, HA As Double

        latrad = degToRad(lat)
```

```
        sdRad = degToRad(SolarDec)

        HAarg = (Cos(degToRad(90.833)) / (Cos(latrad) * Cos(sdRad)) - Tan(latrad) * Tan(sdRad))

        HA = (Application.WorksheetFunction.Acos(Cos(degToRad(90.833)) _
              / (Cos(latrad) * Cos(sdRad)) - Tan(latrad) * Tan(sdRad)))

        calcHourAngleSunrise = HA

End Function

Function calcHourAngleSunset(lat, SolarDec)

'***********************************************************************/
'* Name:    calcHourAngleSunset
'* Type:    Function
'* Purpose: calculate the hour angle of the sun at sunset for the
'*         latitude
'* Arguments:
'*   lat : latitude of observer in degrees
'* solarDec : declination angle of sun in degrees
'* Return value:
'*   hour angle of sunset in radians
'***********************************************************************/

Dim latrad As Double, sdRad As Double, HAarg As Double, HA As Double

        latrad = degToRad(lat)
        sdRad = degToRad(SolarDec)

        HAarg = (Cos(degToRad(90.833)) / (Cos(latrad) * Cos(sdRad)) - Tan(latrad) * Tan(sdRad))

        HA = (Application.WorksheetFunction.Acos(Cos(degToRad(90.833)) _
               / (Cos(latrad) * Cos(sdRad)) - Tan(latrad) * Tan(sdRad)))

        calcHourAngleSunset = -HA

End Function

Function calcSunriseUTC(jd, Latitude, longitude)

'***********************************************************************/
'* Name:    calcSunriseUTC
'* Type:    Function
'* Purpose: calculate the Universal Coordinated Time (UTC) of sunrise
'*         for the given day at the given location on earth
'* Arguments:
'*   JD  : julian day
'*   latitude : latitude of observer in degrees
'*   longitude : longitude of observer in degrees
'* Return value:
'*   time in minutes from zero Z
'***********************************************************************/

Dim t As Double, eqtime As Double, SolarDec As Double, hourangle As Double
Dim delta As Double, timeDiff As Double, timeUTC As Double
Dim newt As Double

        t = calcTimeJulianCent(jd)

'        // *** First pass to approximate sunrise

        eqtime = calcEquationOfTime(t)
        SolarDec = calcSunDeclination(t)
        hourangle = calcHourAngleSunrise(Latitude, SolarDec)

        delta = longitude - radToDeg(hourangle)
        timeDiff = 4 * delta
' in minutes of time
        timeUTC = 720 + timeDiff - eqtime
' in minutes

' *** Second pass includes fractional jday in gamma calc

        newt = calcTimeJulianCent(calcJDFromJulianCent(t) + timeUTC / 1440#)
        eqtime = calcEquationOfTime(newt)
        SolarDec = calcSunDeclination(newt)
        hourangle = calcHourAngleSunrise(Latitude, SolarDec)
        delta = longitude - radToDeg(hourangle)
        timeDiff = 4 * delta
        timeUTC = 720 + timeDiff - eqtime
```

```
' in minutes

        calcSunriseUTC = timeUTC

End Function

Function calcSolNoonUTC(t, longitude)

'***********************************************************************/
'* Name:    calcSolNoonUTC
'* Type:    Function
'* Purpose: calculate the Universal Coordinated Time (UTC) of solar
'*     noon for the given day at the given location on earth
'* Arguments:
'*   t : number of Julian centuries since J2000.0
'*   longitude : longitude of observer in degrees
'* Return value:
'*   time in minutes from zero Z
'***********************************************************************/

Dim newt As Double, eqtime As Double, solarNoonDec As Double, solNoonUTC As Double

        newt = calcTimeJulianCent(calcJDFromJulianCent(t) + 0.5 + longitude / 360#)

        eqtime = calcEquationOfTime(newt)
        solarNoonDec = calcSunDeclination(newt)
        solNoonUTC = 720 + (longitude * 4) - eqtime

        calcSolNoonUTC = solNoonUTC

End Function

Function calcSunsetUTC(jd, Latitude, longitude)

'***********************************************************************/
'* Name:    calcSunsetUTC
'* Type:    Function
'* Purpose: calculate the Universal Coordinated Time (UTC) of sunset
'*         for the given day at the given location on earth
'* Arguments:
'*   JD  : julian day
'*   latitude : latitude of observer in degrees
'*   longitude : longitude of observer in degrees
'* Return value:
'*   time in minutes from zero Z
'***********************************************************************/

Dim t As Double, eqtime As Double, SolarDec As Double, hourangle As Double
Dim delta As Double, timeDiff As Double, timeUTC As Double
Dim newt As Double

        t = calcTimeJulianCent(jd)

'        // First calculates sunrise and approx length of day

        eqtime = calcEquationOfTime(t)
        SolarDec = calcSunDeclination(t)
        hourangle = calcHourAngleSunset(Latitude, SolarDec)

        delta = longitude - radToDeg(hourangle)
        timeDiff = 4 * delta
        timeUTC = 720 + timeDiff - eqtime

'        // first pass used to include fractional day in gamma calc

        newt = calcTimeJulianCent(calcJDFromJulianCent(t) + timeUTC / 1440#)
        eqtime = calcEquationOfTime(newt)
        SolarDec = calcSunDeclination(newt)
        hourangle = calcHourAngleSunset(Latitude, SolarDec)

        delta = longitude - radToDeg(hourangle)
        timeDiff = 4 * delta
        timeUTC = 720 + timeDiff - eqtime
'        // in minutes

        calcSunsetUTC = timeUTC

End Function

Function sunrise(lat, lon, year, month, day, timezone, dlstime)
```

```
'***********************************************************************/
'* Name:    sunrise
'* Type:    Main Function called by spreadsheet
'* Purpose: calculate time of sunrise  for the entered date
'*     and location.
'* For latitudes greater than 72 degrees N and S, calculations are
'* accurate to within 10 minutes. For latitudes less than +/- 72°
'* accuracy is approximately one minute.
'* Arguments:
'   latitude = latitude (decimal degrees)
'   longitude = longitude (decimal degrees)
'    NOTE: longitude is negative for western hemisphere for input cells
'          in the spreadsheet for calls to the functions named
'          sunrise, solarnoon, and sunset. Those functions convert the
'          longitude to positive for the western hemisphere for calls to
'          other functions using the original sign convention
'          from the NOAA javascript code.
'   year = year
'   month = month
'   day = day
'   timezone = time zone hours relative to GMT/UTC (hours)
'   dlstime = daylight savings time (0 = no, 1 = yes) (hours)
'* Return value:
'*   sunrise time in local time (days)
'***********************************************************************/

Dim longitude As Double, Latitude As Double, jd As Double
Dim riseTimeGMT As Double, riseTimeLST As Double

' change sign convention for longitude from negative to positive in western hemisphere
            longitude = lon * -1
            Latitude = lat
            If (Latitude > 89.8) Then Latitude = 89.8
            If (Latitude < -89.8) Then Latitude = -89.8

            jd = calcJD(year, month, day)

'            // Calculate sunrise for this date
            riseTimeGMT = calcSunriseUTC(jd, Latitude, longitude)

'            //  adjust for time zone and daylight savings time in minutes
            riseTimeLST = riseTimeGMT + (60 * timezone) + (dlstime * 60)

'            //  convert to days
            sunrise = riseTimeLST / 1440

End Function

Function solarnoon(lat, lon, year, month, day, timezone, dlstime)

'***********************************************************************/
'* Name:    solarnoon
'* Type:    Main Function called by spreadsheet
'* Purpose: calculate the Universal Coordinated Time (UTC) of solar
'*     noon for the given day at the given location on earth
'* Arguments:
'    year
'    month
'    day
'*   longitude : longitude of observer in degrees
'    NOTE: longitude is negative for western hemisphere for input cells
'          in the spreadsheet for calls to the functions named
'          sunrise, solarnoon, and sunset. Those functions convert the
'          longitude to positive for the western hemisphere for calls to
'          other functions using the original sign convention
'          from the NOAA javascript code.
'* Return value:
'*   time of solar noon in local time days
'***********************************************************************/

Dim longitude As Double, Latitude As Double, jd As Double
Dim t As Double, newt As Double, eqtime As Double
Dim solarNoonDec As Double, solNoonUTC As Double

' change sign convention for longitude from negative to positive in western hemisphere
        longitude = lon * -1
        Latitude = lat
        If (Latitude > 89.8) Then Latitude = 89.8
        If (Latitude < -89.8) Then Latitude = -89.8

        jd = calcJD(year, month, day)
```

```
        t = calcTimeJulianCent(jd)

        newt = calcTimeJulianCent(calcJDFromJulianCent(t) + 0.5 + longitude / 360#)

        eqtime = calcEquationOfTime(newt)
        solarNoonDec = calcSunDeclination(newt)
        solNoonUTC = 720 + (longitude * 4) - eqtime

'            //  adjust for time zone and daylight savings time in minutes
        solarnoon = solNoonUTC + (60 * timezone) + (dlstime * 60)

'            //  convert to days
        solarnoon = solarnoon / 1440

End Function

Function sunset(lat, lon, year, month, day, timezone, dlstime)

'***********************************************************************/
'* Name:    sunset
'* Type:    Main Function called by spreadsheet
'* Purpose: calculate time of sunrise and sunset for the entered date
'*     and location.
'* For latitudes greater than 72 degrees N and S, calculations are
'* accurate to within 10 minutes. For latitudes less than +/- 72°
'* accuracy is approximately one minute.
'* Arguments:
'   latitude = latitude (decimal degrees)
'   longitude = longitude (decimal degrees)
'    NOTE: longitude is negative for western hemisphere for input cells
'          in the spreadsheet for calls to the functions named
'          sunrise, solarnoon, and sunset. Those functions convert the
'          longitude to positive for the western hemisphere for calls to
'          other functions using the original sign convention
'          from the NOAA javascript code.
'   year = year
'   month = month
'   day = day
'   timezone = time zone hours relative to GMT/UTC (hours)
'   dlstime = daylight savings time (0 = no, 1 = yes) (hours)
'* Return value:
'*   sunset time in local time (days)
'***********************************************************************/

Dim longitude As Double, Latitude As Double, jd As Double
Dim setTimeGMT As Double, setTimeLST As Double

' change sign convention for longitude from negative to positive in western hemisphere
            longitude = lon * -1
            Latitude = lat
            If (Latitude > 89.8) Then Latitude = 89.8
            If (Latitude < -89.8) Then Latitude = -89.8

            jd = calcJD(year, month, day)

'           // Calculate sunset for this date
            setTimeGMT = calcSunsetUTC(jd, Latitude, longitude)

'            //  adjust for time zone and daylight savings time in minutes
            setTimeLST = setTimeGMT + (60 * timezone) + (dlstime * 60)

'            //  convert to days
            sunset = setTimeLST / 1440

End Function

Function solarazimuth(lat, lon, year, month, day, _
                      hours, minutes, seconds, timezone, dlstime)

'***********************************************************************/
'* Name:    solarazimuth
'* Type:    Main Function
'* Purpose: calculate solar azimuth (deg from north) for the entered
'*          date, time and location. Returns -999999 if darker than twilight
'*
'* Arguments:
'*   latitude, longitude, year, month, day, hour, minute, second,
'*   timezone, daylightsavingstime
'* Return value:
'*   solar azimuth in degrees from north
'*
```

```
'* Note: solarelevation and solarazimuth functions
'*       could be converted to a VBA subroutine that would return
'*       both values.
'*
'***********************************************************************/

Dim longitude As Double, Latitude As Double
Dim Zone As Double, daySavings As Double
Dim hh As Double, mm As Double, ss As Double, timenow As Double
Dim jd As Double, t As Double, r As Double
Dim alpha As Double, theta As Double, Etime As Double, eqtime As Double
Dim SolarDec As Double, earthRadVec As Double, solarTimeFix As Double
Dim trueSolarTime As Double, hourangle As Double, harad As Double
Dim csz As Double, zenith As Double, azDenom As Double, azRad As Double
Dim azimuth As Double, exoatmElevation As Double
Dim step1 As Double, step2 As Double, step3 As Double
Dim refractionCorrection As Double, Te As Double, solarzen As Double

            longitude = lon * -1
            Latitude = lat
            If (Latitude > 89.8) Then Latitude = 89.8
            If (Latitude < -89.8) Then Latitude = -89.8

            Zone = timezone * -1
            daySavings = dlstime * 60
            hh = hours - (daySavings / 60)
            mm = minutes
            ss = seconds

'//    timenow is GMT time for calculation in hours since 0Z
            timenow = hh + mm / 60 + ss / 3600 + Zone

            jd = calcJD(year, month, day)
            t = calcTimeJulianCent(jd + timenow / 24#)
            r = calcSunRadVector(t)
            alpha = calcSunRtAscension(t)
            theta = calcSunDeclination(t)
            Etime = calcEquationOfTime(t)

            eqtime = Etime
            SolarDec = theta '//    in degrees
            earthRadVec = r

            solarTimeFix = eqtime - 4# * longitude + 60# * Zone
            trueSolarTime = hh * 60# + mm + ss / 60# + solarTimeFix
            '//    in minutes

            Do While (trueSolarTime > 1440)
                trueSolarTime = trueSolarTime - 1440
            Loop

            hourangle = trueSolarTime / 4# - 180#
            '//    Thanks to Louis Schwarzmayr for the next line:
            If (hourangle < -180) Then hourangle = hourangle + 360#

            harad = degToRad(hourangle)

            csz = Sin(degToRad(Latitude)) * _
                  Sin(degToRad(SolarDec)) + _
                  Cos(degToRad(Latitude)) * _
                  Cos(degToRad(SolarDec)) * Cos(harad)

            If (csz > 1#) Then
                csz = 1#
            ElseIf (csz < -1#) Then
                csz = -1#
            End If

            zenith = radToDeg(Application.WorksheetFunction.Acos(csz))

            azDenom = (Cos(degToRad(Latitude)) * Sin(degToRad(zenith)))

            If (Abs(azDenom) > 0.001) Then
                azRad = ((Sin(degToRad(Latitude)) * _
                    Cos(degToRad(zenith))) - _
                    Sin(degToRad(SolarDec))) / azDenom
                If (Abs(azRad) > 1#) Then
                    If (azRad < 0) Then
                        azRad = -1#
                    Else
                        azRad = 1#
```

```
                End If
            End If

            azimuth = 180# - radToDeg(Application.WorksheetFunction.Acos(azRad))

            If (hourangle > 0#) Then
                azimuth = -azimuth
            End If
        Else
            If (Latitude > 0#) Then
                azimuth = 180#
            Else
                azimuth = 0#
            End If
        End If
        If (azimuth < 0#) Then
            azimuth = azimuth + 360#
        End If

        exoatmElevation = 90# - zenith

        If (exoatmElevation > 85#) Then
            refractionCorrection = 0#
        Else
            Te = Tan(degToRad(exoatmElevation))
            If (exoatmElevation > 5#) Then
                refractionCorrection = 58.1 / Te - 0.07 / (Te * Te * Te) + _
                    0.000086 / (Te * Te * Te * Te * Te)
            ElseIf (exoatmElevation > -0.575) Then
                step1 = (-12.79 + exoatmElevation * 0.711)
                step2 = (103.4 + exoatmElevation * (step1))
                step3 = (-518.2 + exoatmElevation * (step2))
                refractionCorrection = 1735# + exoatmElevation * (step3)
            Else
                refractionCorrection = -20.774 / Te
            End If
            refractionCorrection = refractionCorrection / 3600#
        End If

        solarzen = zenith - refractionCorrection
        solarazimuth = azimuth

End Function

Function solarelevation(lat, lon, year, month, day, _
                      hours, minutes, seconds, timezone, dlstime)

'***********************************************************************/
'* Name:    solarazimuth
'* Type:    Main Function
'* Purpose: calculate solar azimuth (deg from north) for the entered
'*          date, time and location. Returns -999999 if darker than twilight
'*
'* Arguments:
'*   latitude, longitude, year, month, day, hour, minute, second,
'*   timezone, daylightsavingstime
'* Return value:
'*   solar azimuth in degrees from north
'*
'* Note: solarelevation and solarazimuth functions
'*       could converted to a VBA subroutine that would return
'*       both values.
'*
'***********************************************************************/

Dim longitude As Double, Latitude As Double
Dim Zone As Double, daySavings As Double
Dim hh As Double, mm As Double, ss As Double, timenow As Double
Dim jd As Double, t As Double, r As Double
Dim alpha As Double, theta As Double, Etime As Double, eqtime As Double
Dim SolarDec As Double, earthRadVec As Double, solarTimeFix As Double
Dim trueSolarTime As Double, hourangle As Double, harad As Double
Dim csz As Double, zenith As Double, azDenom As Double, azRad As Double
Dim azimuth As Double, exoatmElevation As Double
Dim step1 As Double, step2 As Double, step3 As Double
Dim refractionCorrection As Double, Te As Double, solarzen As Double

        longitude = lon * -1
        Latitude = lat
        If (Latitude > 89.8) Then Latitude = 89.8
        If (Latitude < -89.8) Then Latitude = -89.8
```

```
            Zone = timezone * -1
            daySavings = dlstime * 60
            hh = hours - (daySavings / 60)
            mm = minutes
            ss = seconds

'//    timenow is GMT time for calculation in hours since 0Z
            timenow = hh + mm / 60 + ss / 3600 + Zone

            jd = calcJD(year, month, day)
            t = calcTimeJulianCent(jd + timenow / 24#)
            r = calcSunRadVector(t)
            alpha = calcSunRtAscension(t)
            theta = calcSunDeclination(t)
            Etime = calcEquationOfTime(t)

            eqtime = Etime
            SolarDec = theta '//    in degrees
            earthRadVec = r

            solarTimeFix = eqtime - 4# * longitude + 60# * Zone
            trueSolarTime = hh * 60# + mm + ss / 60# + solarTimeFix
            '//    in minutes

            Do While (trueSolarTime > 1440)
                trueSolarTime = trueSolarTime - 1440
            Loop

            hourangle = trueSolarTime / 4# - 180#
            '//    Thanks to Louis Schwarzmayr for the next line:
            If (hourangle < -180) Then hourangle = hourangle + 360#

            harad = degToRad(hourangle)

            csz = Sin(degToRad(Latitude)) * _
                  Sin(degToRad(SolarDec)) + _
                  Cos(degToRad(Latitude)) * _
                  Cos(degToRad(SolarDec)) * Cos(harad)

            If (csz > 1#) Then
                csz = 1#
            ElseIf (csz < -1#) Then
                csz = -1#
            End If

            zenith = radToDeg(Application.WorksheetFunction.Acos(csz))

            azDenom = (Cos(degToRad(Latitude)) * Sin(degToRad(zenith)))

            If (Abs(azDenom) > 0.001) Then
                azRad = ((Sin(degToRad(Latitude)) * _
                    Cos(degToRad(zenith))) - _
                    Sin(degToRad(SolarDec))) / azDenom
                If (Abs(azRad) > 1#) Then
                    If (azRad < 0) Then
                        azRad = -1#
                    Else
                        azRad = 1#
                    End If
                End If

                azimuth = 180# - radToDeg(Application.WorksheetFunction.Acos(azRad))

                If (hourangle > 0#) Then
                    azimuth = -azimuth
                End If
            Else
                If (Latitude > 0#) Then
                    azimuth = 180#
                Else
                    azimuth = 0#
                End If
            End If
            If (azimuth < 0#) Then
                azimuth = azimuth + 360#
            End If

            exoatmElevation = 90# - zenith

            If (exoatmElevation > 85#) Then
```

```
            refractionCorrection = 0#
        Else
            Te = Tan(degToRad(exoatmElevation))
            If (exoatmElevation > 5#) Then
                refractionCorrection = 58.1 / Te - 0.07 / (Te * Te * Te) + _
                    0.000086 / (Te * Te * Te * Te * Te)
            ElseIf (exoatmElevation > -0.575) Then
                step1 = (-12.79 + exoatmElevation * 0.711)
                step2 = (103.4 + exoatmElevation * (step1))
                step3 = (-518.2 + exoatmElevation * (step2))
                refractionCorrection = 1735# + exoatmElevation * (step3)
            Else
                refractionCorrection = -20.774 / Te
            End If
            refractionCorrection = refractionCorrection / 3600#
        End If

        solarzen = zenith - refractionCorrection
        solarelevation = 90# - solarzen

End Function
```

10 APPENDIX C: SEDIMENT-WATER HEAT EXCHANGE

Although the omission of sediment-water heat exchange is usually justified for deeper systems, it can have a significant impact on the heat balance for shallower streams. Consequently, sediment-water heat exchange is included in QUAL2K.

A major impediment to its inclusion is that incorporating sediment heat transfer often carries a heavy computational burden. This is because the sediments are usually represented as a vertically segmented distributed system. Thus, inclusion of the mechanism results in the addition of numerous sediment segments for each overlying water reach.

In the present appendix, I derive a computationally-efficient lumped approach that yields comparable results to the distributed methods.

The conduction equation is typically used to simulate the vertical temperature distribution in a distributed sediment (Figure 56*a*)

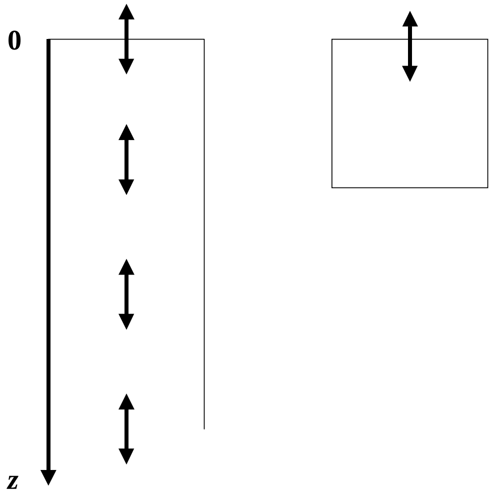
$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \qquad \textbf{(207)}$$

This model can be subjected to the following boundary conditions:

$$T(0,t) = \overline{T} + T_a \cos[\omega(t - \phi)]$$

$$T(\infty,t) = \overline{T}$$

where T = sediment temperature [^{o}C], t = time [s], α = sediment thermal diffusivity [m^2 s^{-1}], and z = depth into the sediments [m], where z = 0 at the sediment-water interface and z increases downward, $\overline{T}$ = mean temperature of overlying water [^{o}C], T_a = amplitude of temperature of overlying water [^{o}C], ω = frequency [s^{-1}] = $2\pi/T_p$, T_p = period [s], and ϕ = phase lag [s]. The first boundary condition specifies a sinusoidal Dirichlet boundary condition at the sediment-water interface. The second specifies a constant temperature at infinite depth. Note that the mean of the surface sinusoid and the lower fixed temperature are identical.



(*a*) distributed **(*b*) Lumped**

Figure 56. Alternate representations of sediments: (*a*) distributed and (*b*) lumped.

Applying these boundary conditions, Equation (177) can be solved for (Carslaw and Jaeger 1959)

$$T(z,t) = \overline{T} + T_a e^{-\omega' z} \cos[\omega(t-\phi) - \omega' z] \tag{208}$$

where ω' [m^{-1}] is defined as

$$\omega' = \sqrt{\frac{\omega}{2\alpha}} \tag{209}$$

The heat flux at the sediment water interface can then be determined by substituting the derivative of Equation (178) into Fourier's law and evaluating the result at the sediment-water interface (z = 0) to yield

$$J(0,t) = \rho C_p \sqrt{\omega\alpha}\, T_a \cos[\omega(t-\phi) + \pi/4] \tag{210}$$

where $J(0, t)$ = flux [W/m^2].

An alternative approach can be developed using a first-order lumped model (Figure 56*b*),

$$H_s \rho_s C_{ps} \frac{dT_s}{dt} = \frac{\alpha_s \rho_s C_{ps}}{H_s/2}\left[\overline{T} + T_a \cos[\omega(t-\phi)] - T_s\right]$$

where H_{sed} = the thickness of the sediment layer [m], ρ_s = sediment density [kg/m^3], and C_{ps} = sediment heat capacity [joule (kg ^{o}C)]$^{-1}$]. Collecting terms gives,

$$\frac{dT}{dt} + k_h T = k_h \overline{T} + k_h T_a \cos[\omega(t-\phi)]$$

where

$$k_h = \frac{2\alpha_s}{H_{sed}{}^2}$$

After initial transient have died out, this solution to this equation is

$$T = \overline{T} + \frac{k_h}{\sqrt{k_h{}^2 + \omega^2}} T_a \cos\left[\omega(t - \phi) - \tan^{-1}(\omega / k_h)\right] \tag{211}$$

which can be used to determine the flux as

$$J = \frac{2\alpha}{H_{sed}} \rho C_p T_a \left[\cos[\omega(t - \phi)] - \frac{k_h}{\sqrt{k_h{}^2 + \omega^2}} \cos\left[\omega(t - \phi) - \tan^{-1}(\omega / k_h)\right] \right] \tag{212}$$

It can be shown that Equations (180) and (182) yield identical results if the depth of the single layer is set at

$$H_{sed} = \frac{1}{\omega'} \tag{213}$$

Water quality models typically consider annual, weekly and diurnal variations. Using $\alpha = 0.0035$ cm^2/s (Hutchinson 1957), the single-layer depth that would capture these frequencies can be calculated as 2.2 m, 30 cm and 12 cm, respectively.

Because QUAL2K resolves diel variations, a value on the order of 12 cm should be selected for the sediment thickness. We have chosen of value of 10 cm as being an adequate first estimate because of the uncertainties of the river sediment thermal properties (Table 4).