
11 DSM2-QUAL Initialization

11.1 Introduction

Formally, physically-based models such as DSM2-HYDRO and DSM2-QUAL require initial and boundary conditions. These models march through time, using boundary conditions and physical conservation laws to update the state of the delta from one time step to the next. The marching proceeds from an initial state, whose influence is gradually lost as the simulation proceeds and the system “loses memory” of its initial condition.

In practical modeling, there are circumstances when initial conditions can be neglected (i.e., specified arbitrarily). When the period to be modeled is long compared to system memory, the initial condition influences only a small fraction of the run. In such cases, the model is often “cold started” from a numerically convenient initial condition. For example, HYDRO is almost “cold started,” because it loses memory of a previous state within a day or two – a period much shorter than that of most modeling applications.

Water quality models such as QUAL have a longer memory of several months, which is not short in context of real-time applications of several weeks. At the end of a 14-day salinity simulation, much of the salt in the interior delta will have originated from somewhere within the model domain. The contribution of the initial salinity field must be accounted for, but this is a technical challenge – because at any moment salinity can only be known, with error, at a few dozen monitoring stations.

This report begins by introducing the Lagrangian mechanics of QUAL and discussing some of the interplay between various sources of error in the model. The subsequent sections survey options available for quality model initialization and introduce an optimization-based method that increases short-medium term accuracy. Finally, three methods are tested, and the results attest to the importance of the initial condition and the inadequacy of methods that do not incorporate field data.

11.2 Transport in QUAL

Figure 11-1 is a rudimentary portrayal of transport mechanics as modeled in QUAL. QUAL is written from a Lagrangian viewpoint with a moving frame of reference – the volume in a channel is subdivided into parcels, and these parcels are tracked as they move together with the local flow velocity. This movement as a “train” of parcels is known as “advection.” When water enters the system, a new parcel is created to

accommodate the inflow and its concentration is taken from the boundary condition. In Figure 11-1, if the water is moving toward the right, Parcel 0 would represent an inflow from a boundary on the left side. When a parcel fully exits the system it is destroyed (Parcel 5 is a candidate if the right side is a boundary). Dispersion and other “mixing” processes are depicted in QUAL as a volume exchange of fluid between neighboring parcels at each time step. Finally, an observer as shown in Figure 11-1 as being fixed and “watching the train cars go by.”

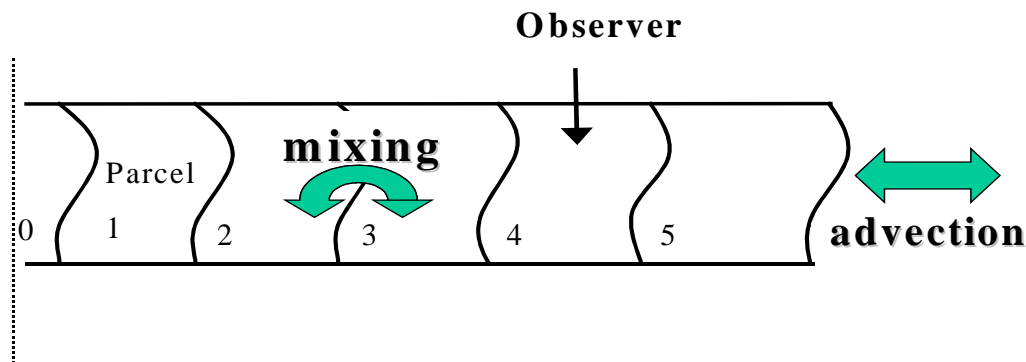


Figure 11-1: Advection and mixing of parcels in a channel.

Neither advection nor mixing is a speedy mechanism for carrying salt inland. Parcels flowing in from the ocean boundary tend to exit during the subsequent ebb or later in the spring-neap cycle. Ordinarily, for salt to persist in the delta, parcels making an incursion into the delta must mix with the water in the interior before leaving. In the simplistic example of Figure 11-1, Parcel 0 might appear during a (rightward) inflow, mix a bit with Parcel 1 and then leave again. Eventually, some of the new salt will disperse toward the right. In the meantime, the observer sees the salt from the interior parcels (perhaps Parcels 3-4) as they oscillate back and forth with the tide.

11.3 Sources of Error

Based on the discussion so far, quality simulation error can be attributed to three sources: initial conditions, boundary conditions, and model error.

11.3.1 Initial Conditions

At any time, the state of the Delta can be estimated by a snapshot of the 20-30 water quality monitoring sites. Estimates of initial conditions will be at gage accuracy near monitoring stations, worse in between. “Gage accuracy” here is meant to encompass instrument accuracy, representativeness of the cross-section, and the deficiencies of using EC as a surrogate for salinity.

The resolution of the initial condition in QUAL is limited. Although it is convenient to talk about initializing parcels (parcels being the fundamental unit of the model), the scale of the initial condition in QUAL is really the channel. One value may be assigned per channel, and is installed in all its parcels – they soon become different as they mix from water from other channels. To the extent that this resolution is adequate, it is because:

1. There are hundreds of channels;
2. The monitoring network is too sparse to justify higher resolution; and
3. Salinity is gradually varying.

The relative importance of initial condition error decreases over time.

11.3.2 Boundary Conditions

Water quality is monitored regularly at all the important boundaries. During historical periods, boundary error depends on “gage accuracy” (as defined above). In the “future” portion of a real-time run, flow and salinity boundary conditions are predicted with ever-decreasing accuracy. Thus, the magnitude of boundary error increases over time unless the run is based purely on historical data.

11.3.3 Model Error

The model itself introduces error. Even if boundary and initial conditions are specified perfectly, the one-dimensional approximation, uncertainty over parameter estimates and other numerical imperfections will distort the solution. This happens slowly.

“Model error” must be distinguished from “model operating on boundary error or on initial error.” As an example of the latter, imagine that the initial salinity has been underestimated and we use the model to simulate the opening of the Delta Cross Channel (such a situation will arise later in the test problem). In such an experiment, the cross-channel opening may cause a milder change because the model mistakenly thinks that salinity is low to begin with. Such a result has a downside and an upside. The downside is that the model underestimates the effect of the delta cross channel operation. The upside is that two wrongs make a right: salinity started out too low and by not decreasing as much will end up closer to the correct value. Such negative feedback is common and as a result, model error tends to be bound in the long run. “Long-term model error” is a concept that is nebulous and problem-dependent, but is nevertheless useful. Long-term model error depends on the quality of calibration (an IEP project is underway to improve long term accuracy, see Chapter 10 of this report).

11.4 Initialization Strategies

The preliminary ideas above can now be applied to survey initialization strategies. Three additional terms will be used to describe the timing of the run (see Figure 11-2):

- t_s will be used to specify the model time at the beginning of the simulation,
- t_0 will be used to specify “now” or the beginning of the prediction phase of the simulation, and
- “spin-up” will refer to the period between t_s and t_0 in which the model warms up (if such a period is required).

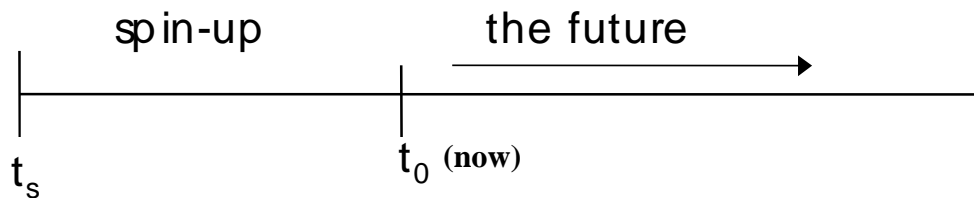


Figure 11-2: Timing of a real time run with spin-up.

11.4.1 Cold Start with Long Spin-Up

A common strategy for real-time applications is to start the model from an arbitrary initial condition and let it spin up for a long period (four-six months of model time is typical). The long initialization period eliminates initial condition error at t_0 , but in exchange for a long period of cumulative model error and boundary error. In the illustration in Figure 11-3, the cold start begins with very large error, because of the arbitrary initial condition. Gradually, the initial error dissipates and the model approaches long-term model error. This is the best that the memory-loss model can do, and there is no recourse at t_0 if the model doesn't match recent observations.

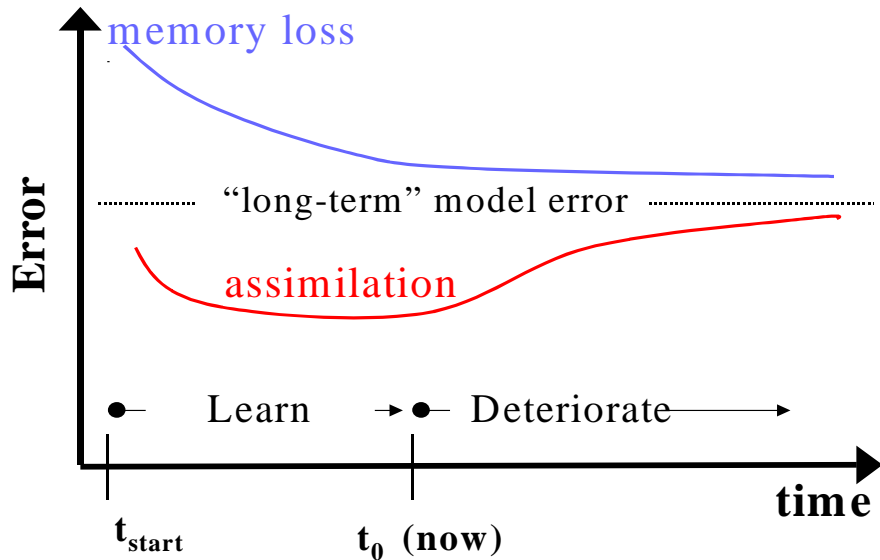


Figure 11-3: Error evolution under “memory loss” and “data assimilation” strategies.

The cold start is attractive because it is simple and skirts the technical difficulties of specifying a network-wide initial condition with limited data. Its biggest drawback is that it is less accurate than other strategies. It can not do better than long term model accuracy. One conclusion of this report is that the cold start is not sufficiently accurate for most real-time applications.

The time-efficiency of the cold start depends on whether the run is performed in isolation or as part of a regular program. A six-month water quality initialization requires not only a six-month water quality run, but also a six-month supporting hydrodynamic simulation to establish a flow field. The hydrodynamic run is a speed bottleneck – hydrodynamic runs of this length currently take a long time to perform (several hours using a 300 MHz machine). If, however, the quality simulation is part of a continuous program of real-time modeling – say once a week – then previous hydrodynamic results can be borrowed and only a modest incremental hydrodynamic run is needed. In this case, the cold start is fast.

11.4.2 Snapshot Initialization

Snapshot initialization is conceptually simple, and corresponds to the intuitive interpretation of initial conditions. The modeler takes the 20-30 or so observations that are available in the Delta at any one instant and interpolates their values over the full network. In snapshot initialization, there is no formal spin-up period, although it is wise to allow a few hours to smooth transient effects at start up. All snapshot methods are vulnerable to station noise and reliability problems because they are based on one value per station.

The snapshot approach is complicated only by the many interpolation schemes that are possible. Here is a brief summary:

11.4.2.1 Patch Schemes

A zero-degree approximation assigns the value at a monitoring station to every channel within a prescribed “area of influence.” Members of the Delta Modeling Section have developed such a scheme.

Figure 11-4 shows the areas of influence ascribed to each monitoring station for a patch schematic based on 31 stations. Note that each patch must contain a monitoring station. The patch-based method is sensitive to the monitoring network size and configuration as well as the assignment of areas of influence – and must be reapportioned when there are missing data. Figure 11-5 shows results for a patch scheme based on 21 stations versus results for 28 stations (from the 31-patch map shown in Figure 11-4, but with 3 stations not available). For the first three weeks, the results are different. Patch-based snapshots of fewer than 20 stations are thought to be inadequate.

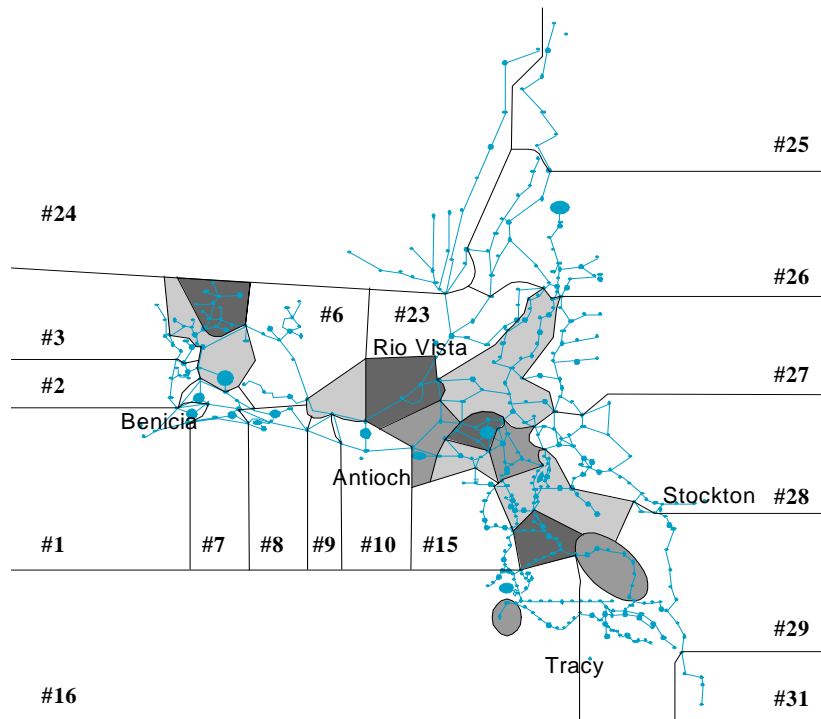


Figure 11-4: Regions of constant initial EC used in the patch-based snapshot scheme.

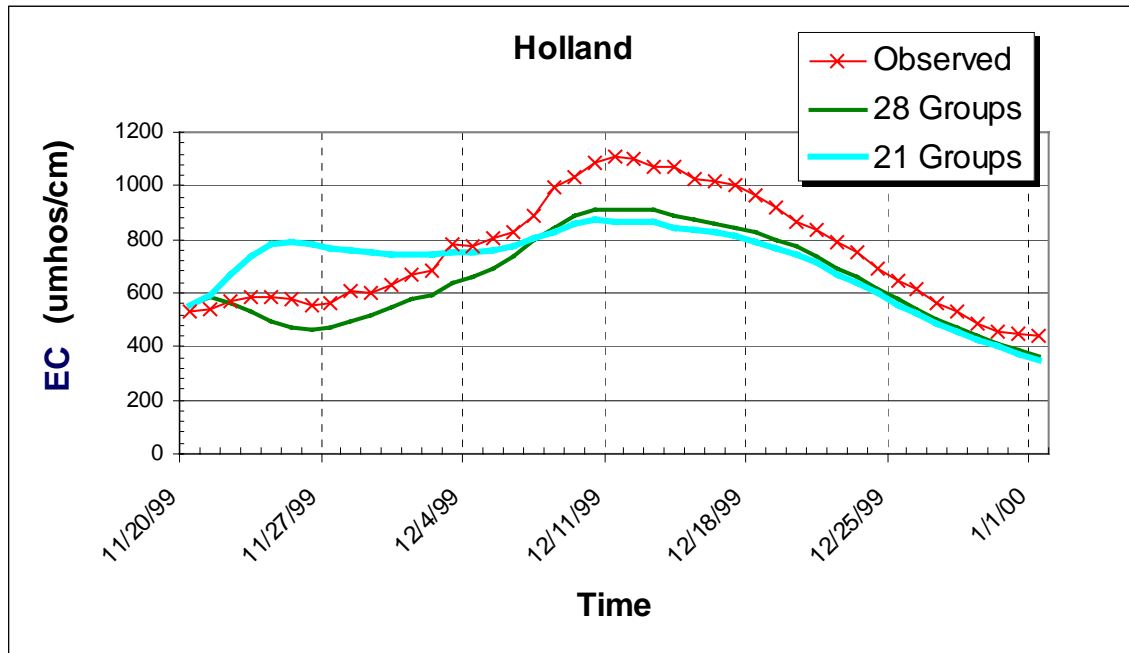


Figure 11-5: EC results and observed data for simulations based on 28 groups and 21 groups.

Because they are based on conditions at one instant, the patch-based snapshot scheme is slightly sensitive to the choice of start time – the snapshot seems to work better at some points in the tidal cycle than others, though this point has not been investigated in detail. This sensitivity to start time was found to be more acute when the patch-based scheme is applied using tidally-averaged EC instead of instantaneous values; for this reason, the use of tidal averages to start the model was abandoned.

11.4.2.2 Higher Order Splines

A smoother initial salinity field between stations can be achieved using higher order approximations (linear interpolation, splines, etc). One difficulty with higher order methods is how to specify them over a network. One proposal by this author is to use a high order monotonic spline to connect stations along the big river systems (Sacramento, San Joaquin, and Old River) that have multiple observations. Linear interpolation would be used in the cross-channels that bridge these main rivers, and zero-order estimates in dead-end sloughs and attached water bodies. Such a scheme cannot be theoretically justified without a sufficiently dense monitoring network, assumptions on the shape of the salinity profile and a high signal-to-noise ratio. The assumption of high signal-to-noise is violated in the South Delta – particularly in the upper San Joaquin where ambient salinity is low but the observation stations pick up spikes of salinity due to agricultural return flows. Higher order schemes were not tested for this paper, but if the noise problem is surmounted they seem likely to compensate for some of the deficiencies of zero-order interpolants.

11.4.2.3 Physical “Smoothing”

As an alternative to filling the salinity field using interpolation, some modelers insert observations into the model using artificial dispersion/diffusion. The advection component of the quality model is turned off, and data are allowed to diffuse in from the observation stations over time, creating a smooth field. No such methods were tested for this paper.

11.4.3 Optimized Initial Condition

So far the snapshot schemes (and the patch schemes in particular) have two main drawbacks. First, the wrong value may be installed between stations if we choose the wrong interpolation method or if stations are too far apart. Second, the schemes are vulnerable to station noise because they use one value per station.

Data assimilation schemes seek to get around these problems by selecting the initial condition that gives the “best” fit to all monitoring data observed over a spin-up period of several days. The number of data used greatly increases: most delta monitoring stations report hourly values, so that the number of observations available in a three day spin-up is about:

$$3 \text{ days} * 24 \text{ observations per day per station} * 30 \text{ stations} = 2,160 \text{ observations.}$$

The use of sequential measurements helps compensate for the spatial sparseness of the monitoring network. For an illustration see Figure 11-6. Assume that the parcels are initialized at time t using data from Stations 1 and 2 for Parcels 4 and 1 respectively, and that some interpolation scheme is used in between (assume that parcels can be initialized individually). At time $(t+1)$ all the parcels have moved to the right and some mixing has occurred. Parcel 3 now resides at the monitoring station. If a new observation is available at Station 1 at time $(t+1)$, a discrepancy will be observed between model and data, and we will learn something about what should have been put in Parcel 3 in the first place. If we continue to fit data over several tide cycles and observe many parcels, we may improve our initial estimates for all the parcels in the channel. The improvement will be largest for parcels whose tidal excursion brings them under the monitoring station soon after the start of the run; however, since these parcels mix with the others (and are influenced by them), we can indirectly learn something about all of them. So far, no mention has been made of the optimization algorithm or implementation details. These are discussed in detail in section 11.5.

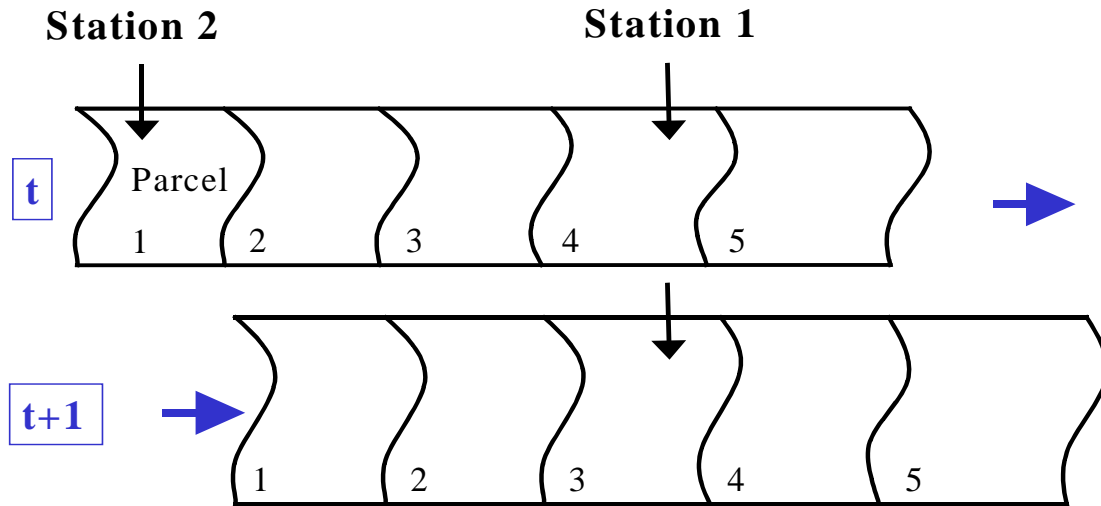


Figure 11-6: Exploiting sequential measurements to gather spatial information.

The evolution of error in a data assimilation scheme is pictured conceptually in Figure 11-3– the scheme is characterized by a period of learning leading up to t_0 followed by a period of deterioration as the model slowly approaches “long-term model error.” Note that Figure 11-3 contains an abuse of scale: the spin-up period (the distance between t_s and t_0) is about 6 months for the memory-loss scheme and about 3 days for assimilation schemes.

The optimization-based method is much faster than a “cold start” in isolation, but slower than a “cold start” that is part of a regular real-time program (see the discussion of “cold start” efficiency above). The optimization-based method is slower than snapshot methods, although it saves on some of the decisions and experimentation required in timing the “snapshot.” The optimization component is global and well behaved.

11.4.4 Recursive data assimilation

One more way to take advantage of new observations is to assimilate data using a recursive filter (Kalman filter). In the example from Figure 11-6, we would correct the parcels on the spot at time $(t+1)$ and send them on their way, instead of going back and trying to find a better initial condition. The Kalman filter does this, nudging the model towards the data recursively. The Kalman filter is based on the premise that both model and data contain error, and the relative magnitude determines the extent of the “nudge.”

Recursive filtering trivializes the initial condition – the idea is not to find a good initial condition at all, but rather to run the simulation continuously, gradually folding in new data. Thus, a recursive filter may be the approach most deserving of the label “real-time tool.” Unfortunately, there are several disadvantages:

- ❑ The filter is invasive to the model. The Kalman filter requires computation inside QUAL (derivatives of all the computational steps). This is particularly difficult because of the use of sub-time steps.
- ❑ Standard Kalman filters are not appropriate for a Lagrangian problem. QUAL has a variable number of model states (depending on how many parcels are created and destroyed at each time step). The recently developed Kalman filters for rank-deficient “descriptor” systems can be used to address this problem, but the filters must be coded from scratch.
- ❑ The Kalman filter is a recursive computational tool, not a complete model. The approach requires an error model for both observations and QUAL, which would have to be described and estimated from scratch.

Despite the disadvantages, the Kalman filter (perhaps in concert with other startup methods) may eventually prove to be a valuable tool.

11.5 Initial Condition Optimization

11.5.1 Introduction and Formulation

A conceptual picture of initial condition optimization through data assimilation was given in previous sections. The optimization of initial conditions is not a new idea – it can be thought of as a parameter estimation problem and treated using numerical optimal control or adjoint data assimilation algorithms. In light of some of the specifics of QUAL, these methods are not as efficient as the technique advocated here based on the principle of superposition (and related to transfer functions).

Given a flow field, transport is governed by a set of partial differential equations (PDE) which are linear. This means that the principle of superposition applies – see Fisher (1972) or any book on linear systems. The superposition principle says that the sum of any two solutions of the PDE (or scalar multiples of these solutions) will also satisfy the PDE. It is a property of the underlying equations, not a model representation – although it is well preserved by QUAL.

Superposition is illustrated in Figure 11-7. Assume that we have a channel with an initial condition in two patches, one with a concentration of 3 “units,” and the other with a concentration of 2 (top of figure). A known boundary condition $B(t)$ is imposed at both ends of the channel for all time. We observe the evolution of concentration at points throughout the channel. The concentration fields thus observed can be obtained from the following sub-problems:

- one problem with the same boundary conditions as the original problem, but an initial condition of zero everywhere;
- one problem with zero boundary conditions, a nominal concentration of 1 in the first patch and a concentration of zero in the second patch; and
- one problem with zero boundary conditions, a concentration of zero in the first patch and a nominal concentration of 1 in the second patch.

To obtain the solution to the original problem, we sum the first problem, 3.0 times the second problem and 2.0 times the third problem.

Now, turn the problem around, and assume that the initial concentrations of the two patches are not known. From the foregoing discussion, the three fundamental problems above can be combined linearly to construct the solution to any initial value problem involving the same two patches and boundary conditions. Let us seek the linear combination that gives the best fit (say, in a least-squares sense) to the observations, which are available in various places in the channel. Regardless of whether the true initial condition is really in two patches, this is the optimal “two patch” initial condition.

The optimization problem is as follows, written in terms of n patches instead of two:

$$\min_{\alpha_i} \sum_{x,t} w_x \left[C(x,t) - \hat{C}(x,t) \right]^2 \quad (0.1)$$

such that

$$\hat{C}(x,t) = \hat{C}_b(x,t) + \sum_{i=1}^n \alpha_i \hat{C}_i(x,t) \quad (0.2)$$

$$\alpha_i > 0 \quad i = 1..n \quad (0.3)$$

where:

- $C(x,t)$ and $\hat{C}(x,t)$ are respectively the observed and modeled salinity concentrations at discrete station locations (x) and time steps (t),
- $\hat{C}_b(x,t)$ is the salinity concentration field obtained from the sub-problem with boundary conditions and zero initial conditions,
- $\hat{C}_i(x,t)$ is the salinity concentration field obtained from the sub-problem with zero boundary conditions and a nominal initial concentration of unity in patch (i),
- w_i is a weight depending on station scaling (unity in the experiments presented in this paper),
- α_i is the initial condition in the i th patch, or, alternatively, the linear coefficient of the i th basic sub-problem. This is constrained to be positive because it represents a positive physical quantity.

Equations (0.1) through (0.3) comprise a convex linear-quadratic program, a type of problem for which there are reliable solution algorithms (routine QPROG from the IMSL library was used for the prototype). The problem has a single, global minimum. Time/station combinations with missing data can simply be omitted from the fit.

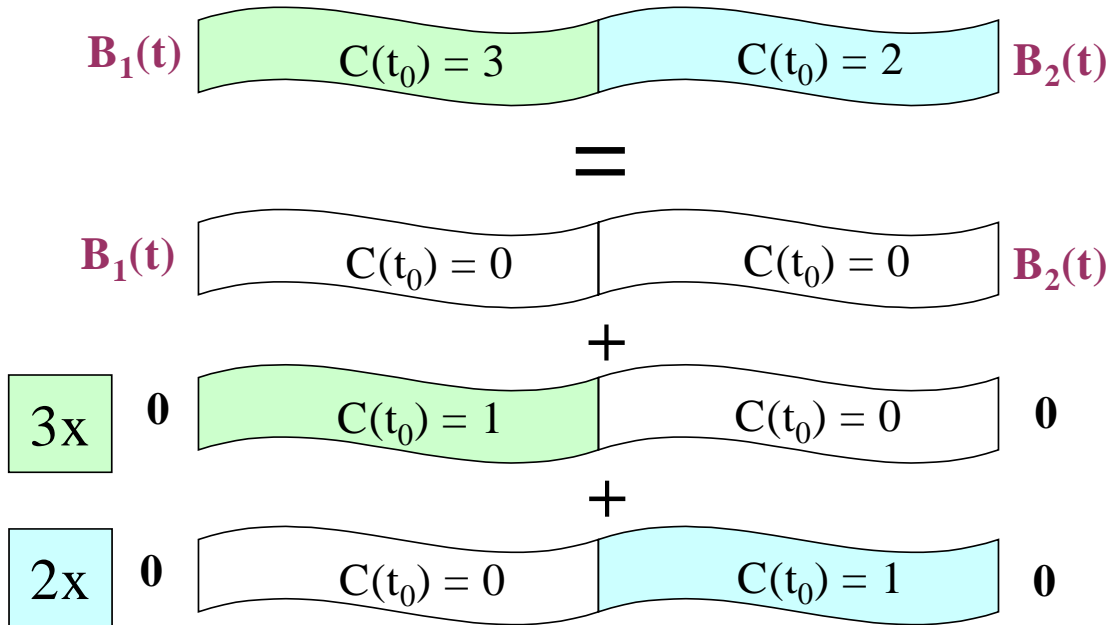


Figure 11-7: Example of superposition. The top initial value problem has been decomposed into a linear combination of simpler sub-problems.

11.5.2 Monotonicity Constraints

The main stems of the Sacramento and the San Joaquin Rivers tend to have salinity that decreases monotonically from the ocean boundary. One-dimensional transport dynamics will always yield monotonic concentration gradients (in the long run) when net flow is toward the ocean boundary and the ocean boundary is the contributor of salt. Such conditions exist along most of the Sacramento, and usually along the San Joaquin to about Jersey Point (upstream on the San Joaquin, the network of channels allows circulatory flow, and salinity spikes appear due to agricultural return flow). In the regions where monotonicity is appropriate, it may be enforced by inequality relationships between neighboring patches:

$$C_i < C_j \quad (i, j) \in M \quad (0.4)$$

where the elements of set M are pairs of patches with a monotonic relationship.

11.5.3 Stability Constraints

The optimization method works best in regions that are well monitored or have a healthy advective exchange with a monitoring station. In small remote areas, the concentration may be less “knowable” and subject to erratic values under optimization. The remote regions may either be lumped together with neighboring patches that are better monitored or they may be bound to these “main” patches by not allowing more than a specified fraction of difference (they can also be fixed *a priori* – see Modifications and Extensions below). A stability constraint may be written:

$$\text{abs}(C_i - C_j) < k C_i \quad (i, j) \in N \quad (0.5)$$

where the elements of N are pairs of “main” (i) and “bound” (j) patches and k is the maximum fraction that the “bound” channel is allowed to deviate from the “main” channel. In practice, this expression is decomposed into two linear constraints.

In the original implementation of the optimization scheme, stability constraints were used to bind neighboring channels, reservoirs to nearby channels (reservoirs are treated just like channels in this scheme) and dead end sloughs to main channels, using $k = 0.2$. In regions that are adequately monitored, stability constraints should be left out so that reservoirs and sloughs can act in their natural role as buffers against salinity change.

11.5.4 Formulation Flexibility

Under the optimization method suggested here, the number of QUAL runs is fixed – once the basic sub-problems have been assembled, the constraints or weights of the optimization problem can be re-specified without additional model runs. The optimization problem itself takes only a fraction of a second to solve.

11.5.5 Efficient Computation and Choosing Patch Size

Under the optimization scheme, we are no longer required to define one patch per monitoring station. In fact, we are at liberty to define “patches” up to the finest resolution available for a QUAL initial condition: one patch per channel.

In practice, we would never want to do this, because it would require hundreds of “fundamental” solutions and the initial condition would be over-resolved compared to the monitoring network. In the schematic used for the tests in this report, there are 82 patches, including 68 channels and 14 reservoirs (which can be treated the same as patches). The areas near Martinez are highly resolved (the salinity gradient is steep and well monitored), but the patches in remote regions in the south delta are lumped in much bigger groups.

There are still 82 patches, and currently performing 82 basic QUAL runs would take a long time – even though the hydrodynamics portion is only run once and the QUAL runs

are conducted only over a 3-day spin up period. Fortunately, it is possible to use QUAL's multiple constituent capability to drastically reduce the number of runs required. "Pseudo-constituents" can be used to represent EC originating from different patches (this idea is also used in "source fingerprinting" studies done by the section), so that about ten patches can be simulated at once. The total computational burden is one three day HYDRO run and about 8-10 QUAL runs. This amount of computation is comparable to that of other methods, and on a 300 MHz machine the method takes about 20 minutes.

11.5.6 Modifications and Extensions

11.5.6.1 Basic Solutions

The methods discussed above use linear combinations of patches for initial conditions. However, we are at liberty to construct solutions from other fundamental building blocks. The only restriction (due to superposition) is that each "building block" can only be adjusted using a scalar multiple.

Instead of separating the basic sub-problems into "boundary" and "patch" constituents, we may use the more general idea of "fixed" and "optimized" regions. The initial condition does not have to be unknown in every region. In some cases, the modeler may want to stipulate salinity *a priori* over part of the model domain. A run using the "fixed" region then becomes analogous to what was previously the "boundary only" run.

Similarly, we can incorporate results from a previous model run to assist in regions that are poorly monitored. The output from a previous run would be part of the "fixed" part of the solution. This solution would be adjusted locally in an amount of detail proportional to the density of the monitoring network. This would mean adjusting each channel near the Western boundary, but adjusting crudely or not at all in remote dead-end sloughs.

11.5.6.2 Filtered Values

It is also possible to use filtered salinity for the optimization instead of instantaneous values. The motivation for using filtered values is to take advantage of the fact that QUAL is more accurate in a tidally-averaged sense than in an instantaneous sense (because of the averaging of phase-dependent errors). Since such averaging is just a type of linear filter, superposition still applies to the output and the optimization method can be applied directly. If averages are used, information loss must be prevented. This requires the use of an appropriate low-pass filtration of the salinity data whose output is defined at the same hourly time step as the original series (rather than some sort of daily average with only one output value defined per day).

11.5.6.3 Monitoring Network

Finally, the results from the optimization method provide a tool for assessing how good the water quality monitoring network is for purposes of model initialization. The optimization method works well in regions that are monitored directly and in regions that are not but have a high advective exchange with a location that is well monitored. By assimilating data over time, the technique “leverages” the information that is available – and its failures indicate locations that require better gage coverage.

11.6 Comparison

A test was conducted between three of the techniques discussed in this report: the cold start, the patch-based snapshot scheme, and the optimization method developed in this report. The test period was November 20 - December 10, 1999, a period of high salinity and dynamic operation, the discussion of which focused attention on real time modeling (it was the subject of the January 2000 Bay Delta Modeling Forum workshop). The Delta Cross Channel was closed on November 26, 1999.

The experiment was conducted with historical boundary flows, in order to control for the influence of boundary error. The cold start spin-up began on February 10, 1999. The snapshot of the “patches” was taken just after midnight on November 20. The optimization spin-up period was November 20-23 with the first four hours ignored to allow the development of a smooth salinity gradient (refining the length of the optimization period is an important item for further development).

Figure 11-8 and Figure 11-9 compare tidally-averaged EC results from the three initialization methods with field data at 6 monitoring stations. Going into the period of interest, the cold start uniformly underestimates salinity at all the stations, sometimes by as much as 50 percent. At some stations, the reaction to the Delta Cross Channel and other operations is accurate despite a persistent negative bias; for instance, at Jersey Point, the shape of the curve matches the shape of the observed curve even though it is way too low. At other stations, the effect of the Delta Cross Channel opening has been attenuated. This is the situation described above where a salinity-controlling operation doesn't do as much because salinity is already low at the outset. The snapshot and optimized methods dramatically reduce bias.

To compare the patch-snapshot method to the optimization-based method, we must neglect the spin-up period November 20-23. After this period, the optimization method yields a further 4-10 days of reduced error compared to the snapshot method at most stations. The reduction in error is most striking at Jersey Point and Rio Vista – but for different reasons. At Jersey Point, the setting is perfect for optimization. The surrounding region is monitored adequately, but the optimization method can help fill in unknown areas. In addition, Jersey Point is remote from the boundary – so the initial condition is critical.

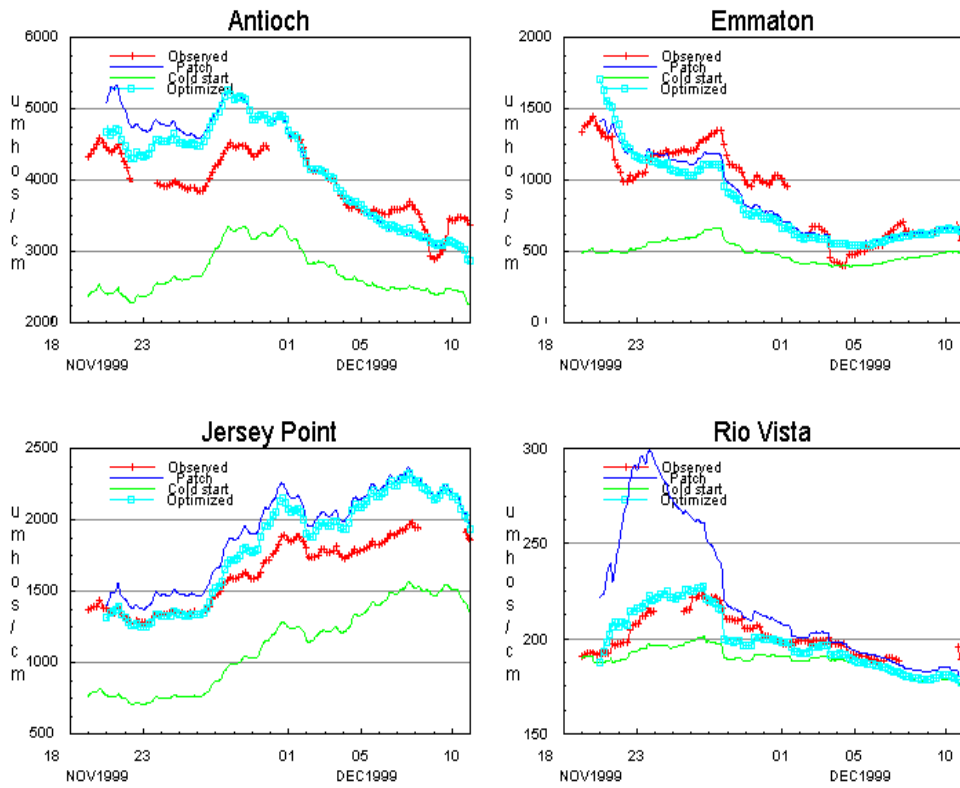


Figure 11-8: Tidally averaged EC for the three methods of model initialization.

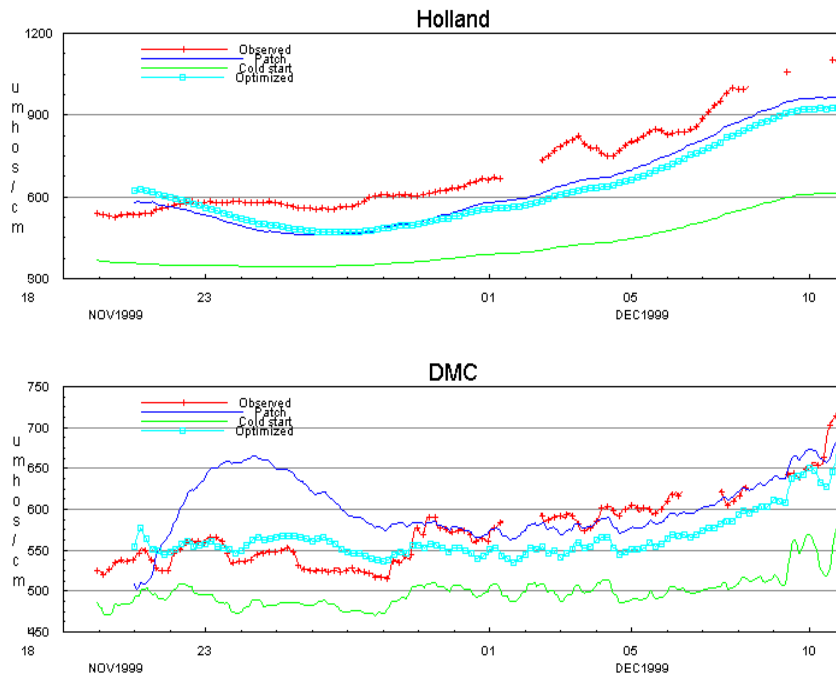


Figure 11-9: Comparison of tidally averaged EC at Holland Tract and DMC.

At Rio Vista (and also DMC), the difference between the two methods is because of a delayed period of overestimation using the patch-snapshot method. This is apparently a hazard of its coarse, zero-degree approximation. In the experiment, a downstream patch with high initial salinity advected towards the Rio Vista area with lower initial salinity, causing an abrupt increase (although the reader is encouraged to look at the scale of the Rio Vista plot). Had the initial model salinity varied more smoothly from high to low, the water that reached Rio Vista would have been less saline and the transition would have been gradual. The optimization-based method achieves smoothness through high resolution. Since in the optimization method the initial patches do not have to contain a monitoring station, many patches can be used (one or two channels per patch was used in this region).

The results at Emmaton are interesting because the patch-snapshot method performed better than the optimization method did, even during the November 20-23 optimization period. Emmaton was partially “sacrificed” in the optimization process in order to reduce error at other stations. This is the only location that the author is aware of where this happened. One reason why this might occur is that station magnitudes vary quite a bit in this region, and unweighted least squared error was used in the objective function. Unweighted least squares preferentially accommodates stations with large magnitudes of error, and Emmaton generally has an EC (and error) almost an order of magnitude lower than that of its western neighbors such as Antioch. The production version of the optimizer has since been rewritten to allow station weighting.

Finally, the salinity trajectories from the snapshot-patch scheme and the optimization-based scheme tend to converge by the end of the test period, particularly in the north (remoteness of the boundary affects convergence). Their mutual approach to the cold start solution, on the other hand, seems to have barely begun. Evidently, “independence of initial conditions” is a relative concept – it takes months for two runs with very distinct initial conditions to converge, but only 10-20 days for two runs with fairly similar initial conditions to do so. It seems safe to assume that runs based on any of the various data-based initialization strategies surveyed in this report (i.e., all of the techniques except the cold start) will be nearly indistinguishable after the third week of simulation.

11.7 Conclusions

This report has surveyed methods of initializing real-time model runs. The most important points are as follows:

- Error arises from a variety of sources. Modelers should be attentive to the “weak point” of the modeling process at various stages of a run. In real-time salinity modeling, initial condition error is an important “weak point,” but one that can be ameliorated.
- It takes a fairly long time for DSM2 to reach its “long-term” level of error – much longer than the duration of a real-time run. When the model is initialized well,

memory works to our advantage and the model deteriorates to “long-term” levels slowly. When the model is initialized poorly, the initial condition will contribute to error for the whole run.

- The optimization-based method described in this article uses a stream of data over time at each station to arrive at a satisfactory initial condition. The scheme plays to QUAL’s strengths, including preservation of superposition and multi-constituent capabilities.

- Alternative tools are available. The rudimentary patch-based snapshot scheme works well considering its simplicity. Recursive filtering is the most conceptually pleasing “real time” tool, but is cumbersome to implement in QUAL.

- The “cold start” (reliance on long-term memory loss) is not sufficiently accurate for real-time applications.

11.8 Reference

Fischer, H.B., E.J. List, R.C.Y. Koh, J. Imberger, and N.H. Brooks. (1972). *Mixing in Inland and Coastal Waters*. Academic Press, San Diego.