BACKING UP OCEAN MACHINE:

I used sudo rsync -azvv -e ssh root@ocean.uoregon.edu:/home/ .

to put it on my local hard drive in /ocean (on OceanBackup disk). After this initial run, try using sudo rsync -auzvv -e ssh root@ocean.uoregon.edu:/home/ . where we added –u option to update only. (from https://help.ubuntu.com/community/BackupYourSystem)

---------------------------------------------

**FVCOM notes**

**5/25/2012**

Testing 2.7.1 with gfortran and netcdf 3.6 on ocean.uoregon.edu

To compile, you have to change the makefile and the casename\_run.dat files to match up and to match what your machine has. See FVCOM manual for more.

Got tst\_run.dat case to work both with and without netcdf I/O. However, to get netcdf output to work I had to modify both mod\_lag.F and mod\_lag.f90 according to help here: http://foros.dgeo.udec.cl/fvcom/viewtopic.php?f=5&t=73

where you change ‘rho1’ to ‘r1’ and get rid of user\_defined. You also have to change scal\_choice to 5 from 6 because you reduced the number of variables.

Also, you have to edit the casename\_run.dat file, at least while using gfortran, to get rid of // characters which are supposed to be line continuations but don’t seem to work. Thus, you edit the K\_Specify and M\_Specify in the Dye release section to make these all fit on one line.

The netcdf file has output in it after running this test case. Now on local machine too to play with in matlab.

Channel test run—also ran successfully with netcdf output. Results now in local folder.

**5/28/12**

Worked on make\_coos.m, altered to make\_coos\_das.m which now works in making all the input files for the Coos Bay run that Geoff Cowles set up for me (i.e,, Google coastline, bathy from DEM, generic rivers, tides from Pt Orford, etc).

Now need GOTM to work. Download from website and build following instructions at http://www.gotm.net/index.php?go=software&page=installation

Now need to check if FVCOM 3.1 works.—used make.inc file that Geoff sent as an example.

Everything straightforward except what the DTLIBS are? Where is fjulian.inc? Is it the problem I am having with mod\_time in the compile??

🡪 UPDATE: Got past mod\_time in compile by getting Julian\_toolkit and compiling.

Now, get error: Fatal Error: Can't open module file 'ocpcomm4.mod' for reading at (1): No such file or directory in mod\_utils.f90 step.

🡪 these file were made when I compiled SWAN, now figuring out how to make it all work…got past this by copying OCPCOMM#.mod to source directory after making clean (otherwise it erases them…).

Getting lots of WARNINGS in compile: Duplicate SAVE attribute specified at (1) for all sorts of variables.

**6/1/12**

Got Intel Fortran and C,C++ compilers. Re-made netcdf-3.6.3 with these and am now making GOTM. Seems to be working- YES! Made executable gotm\_prod\_IFORT.

**6/4/12**

To get Geoff’s FVCOM setup for Coos Bay to run, I had to delete the upper right OBC node because that grid cell had 1 open boundary and 1 solid boundary which wasn’t good for FVCOM apparently. I also had to make sure to write some variables as integers and some as float, in make\_coos\_DAS.m

To run fvcom, first soft link in run directory to the executable

ln –sf ../FVCOM31\_source/fvcom fvcom

then, make sure all input files in input folder, make output directory, and type

./fvcom --casename=coos01 --logfile=log &

where casename matches the input files, and logfile creates log.

I haven’t looked at the output, but I stopped it after about 1.5 days of model time. Saved in test3 in FVCOM/COOS\_BAY/Run directory.

NOTE: I got openMPI 1.6 installed and working on ocean and tested a short hello/world type program that compiled and ran with 32 processes. Though I am not sure if there are 32 processors. There are 0-31 listed in /proc/cpuinfo, so that makes me think there are.

**6/5/12**

Try and compile with parallel activated—uncomment flag 4 in make.inc (saved previous make.inc as make.inc\_serial). Also compiled metis libraries in FVCOM31\_source/libs/metis and point to that under the multiprocessor flag.

make clean; rm fvcom (old executable)

cp ../swan4085/ocp\*.mod .

make

-- didn’t work! evaluate….--> problem was I didn’t change the FC to mpif90 and it was still trying to use ifort. Changed this and it is compiling…(remember to copy over ocpcomm#.mod files from ../swan directory after make clean—see step above).

New problem—gets far in compilation then stops at ice\_therm\_vertical.f90 and says ifort: error #10273: Fatal error in /opt/intel/composer\_xe\_2011\_sp1.10.319/bin/intel64/fortcom, terminated by 0x2

--fix—

comment out –check all in DEBFLAGS definition. Then it gets to end but got an error on –stacksize, which must come from METIS definition in multiprocessor section. Checking this now—drop stack size flag and it compiles.

Ran:

mpirun -np 8 ./fvcom --casename=coos01 --logfile=log &

and it’s going! The Secs/IT time in the logfile is much reduced, so maybe it’s working?? In fact, it is 9.5 times faster! Not sure how that happened, as I am only using 8 processors. Next try with 16, then all 32.

Serial = Secs/IT = 0.3699

-np 8 = Secs/IT = 0.0389

-np 16 = Secs/IT = 0.0207

-np 32 = Secs/IT = 0.0141—all over the map, up 0.0196, 0.0181 just for a few hours model time, then slowly went back down to 0.0142. weird.

\*difference between serial and parallel might be in optimizations etc.

In the meantime, I talked with Craig Rasmussen at CAS IT about using aciss.uoregon.edu, which UO’s supercomputer. I am working with him to get FVCOM up and running there. I gave him a copy of FVCOM31\_source to look at.

**6/6/12**

Decided to test ROMS upwelling case using mpirun and same set of np to see its scaleability. Note—change Ntile variable in ocean\_upwelling.in to match # of processors using. Also, switched back to original 1440 time steps. Will have to redo serial run, but for now test:

mpirun –np 8 oceanM ROMS/External/ocean\_upwelling.in > log &

(for serial runs: ./oceanS < ROMS/External/ocean\_upwelling.in > log &)

-np 8: time = 79.6 seconds (per CPU, for a total = 636 seconds)

-np 16: time = 52 seconds (per CPU, for a total = 832 seconds)

-np 32: time = 46 seconds (per CPU, for a total = 1472 seconds)

Serial: time = 434 secomds

\*\*Note Geoff Cowles commented that small model domains don’t scale as well, and short model runs also mean less of a difference as there is a lot of set up time in the grid partitioning, etc. But at least it all works!

**11/16/12**

Started working on updating Coos Bay grid, both in terms of a new coastline and new bathymetry. The coastline is cobbled together from Sarah Harbert’s GIS work on the OR Coastal Atlas and Geoff’s download of Google Earth data. These are now together in the file /Users/daves/Documents/Coos\_Bay/Bathy\_data/CBay\_coasts\_segmented.mat

which has lon/lat vectors for each section of coast, open bndy, and islands. In addition, the bathymetry now comes from 2 sources: 1) pnw\_combined\_coosbay which is a product mostly of smith&sandwell taken from MoSSea input data and 2) bath\_mllw\_usace\_30m\_withGRID.mat that is built from the Port Orford 1/3 arc second DEM with USACE tracks added on top (to fix the channel not being there near the town of Coos Bay).

To make this new grid, run make\_coos\_new.m in the Model\_Maker directory, pointing it to the right files. The new grid has 18120 vertices and 33486 elements (the older grid above had 4872 vertices and 8304 elements), which is mainly a result of the new grid covering a larger spatial domain, and going from 2 km at the open bndy to 100 m inside Coos Bay.

For now—keep everything else the same, i.e. elevation forcing file, z0 file, river file all same. Only thing I changed is in the coos01\_rivers.nml file I changed the grid\_location to correspond to the node index of where that river is input. I will need to update this file more and go through this river forcing more once I decide to add more rivers.

NOW: upload these new input files to ocean and try to run this! Potential problems might be bad mesh, bad rivers, etc…

-🡪 new grid had a similar issue to before where the lower right OBC node created a problem, so I had to manually make it a solid boundary (now in make\_coos\_new.m)

But then it starts running with command

>> mpirun -np 8 ./fvcom --casename=coos01 --logfile=log &

…and it’s still running, but slower obviously then other model, with 0.18 secs/IT for 8 processors.

- nan’s all over the place at second time step—looking into grid first and trying to learn how to use SMS now. look at coosbay\_xy.dat and readCST.f (where compiled with gfortran readCST.f –o readCST.out)

NOTE: lowered time step to 0.5 seconds (from 1.0 seconds) and it ran successfully!

**11/20/12**

Got SMS grid to work and wrote new make\_coos\_SMS.m to make all other files. However, when I try to run it, the free surface goes off to very high numbers and the river flux (low S) doesn’t make it very far out of the Coos River. I think this means I need to deepen that river channel artificially so it doesn’t get stuck back there and just build up…

---made depths in that channel =2.5 m, but still free surface goes up very fast, so it must be other spots too…

**2/7/2013**

Met with Jim Lerczak and his student, Emily Lemagie, at OSU. They had similar problem to one described above when their river node/channel was only 1 element. When they artificially expanded the end of their river channels, this went away—Emily says:

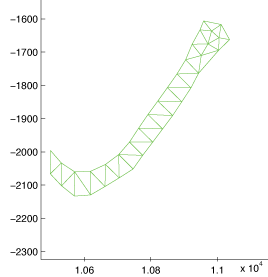
*“When I was constructing the river end of the domain, I originally had a narrow channel that ended at one node. With the initial shape, the model crashed regardless of whether I input the river flow across one or two nodes and regardless of which node I input the freshwater, the water level at the end node would blow up. One alternative is putting the freshwater in at the cell instead of the node (just change the definition for your freshwater inputs), but I ended up shifting the geometry at the end of the river slightly and putting the freshwater in at the end node point. The way our domain is, I don't think this has a significant affect on the main area of interest, but if you have this problem at a lot of input locations you might also try the cell method to see if that changes the stability.”*

With figures:

original river end:



edited river end:



So, I edited the Coos River end where I put in freshwater right now to have similar shape to Emily’s above, with 3 elements, etc🡪 saved in coosbay\_SMS\_Mobj\_riveredit.mat

- also changed to only 1 river node; change .nml file for rivers and for main run to reflect this.

Upload new forcing files to ocean and try and run:

mpirun -np 8 ./fvcom --casename=coosSMS --logfile=log &

in the folder coosSMS\_riveredit on the ocean cluster. It started at least…

-🡪 the river worked after this, but the model blew up after about 2-2.5 days, getting bands of nan’s in salinity, though it was also in free surface, etc.

run again with smoothing bathy by 0.5 this time instead of 0.25—still blew up!

Next: reduce timestep from 1 sec to 0.5 sec and see what happens. Running now on 16 cpu’s: mpirun -np 16 ./fvcom --casename=coosSMS --logfile=log &

Not blowing up yet, but the u/v velocities are going over 10 m/s! Need to check where this is happening and probably fix bathy there…and it did blow up finally.

First NaN’s at 3 days 6 hours in (! 54800 2008-01-04T04:06:40.000000) but the velocities were huge for a day or so before this point…

V-plots: in looking at velocity, there are some strong v’s inside the estuary near some bathy constrictions, but the largest (> 3 m/s) start up in NE corner of BC about 40-48 hours into run…need to check this BC and elevation file? sponge layer?

**2/9/13**

If it is a boundary problem, try a couple different runs to diagnose the problem…

1) Use all OBC nodes (for now, I’ve been not using the corner ones where the OB meets the land, turn these on in make\_coos\_SMS.m), i.e., change nn=2:51 to nn=1:52

- running this case now: still blew up!

2) Turn off elevation file and forcing of tides at OBC, turned off OBC\_ON and OBC\_ELEVATION\_FORCING in the run file coosSMS\_run.nml

- running this case now: this run made it past point where others blew up, so it’s either the forcing file at the BC or the tidal currents it’s inducing are messing things up…, save this log file and output file in coosbaySMS\_noOBC folder in Run folder

- one thought is that the coast goes up to 0 depth at these bndy corners, so maybe deepen the coast at these corners and into the model domain so that elevation file isn’t putting in water into 0 depth

3) Turn back on OBC, but make elevation forcing file all 0’s. This ran for 4 days with no problems as well, so not a file input problem, but something with forcing of tides…

4) Now made vertical wall on coastal bndy outside the bay at 4 m depth; also made maximum depth above mllw = 1.5 (i.e., -1.5 in model); try this with original elevation file; -🡪 this one blew up at ! 24500 2008-01-03T20:03:20.000000, which is faster than before, unfortunately. So maybe it’s the bathy inside the bay…?

5) Next test- make min depth in bay 1.5 m below MLLW, i.e., no dry cells to start with. use original OBC and everything else though: still blew up! at ! 25700 2008-01-03T23:23:20.000000, with vvel’s > 8 m/s.

--not sure what to try now, maybe ask Emily at OSU for tips or her parameters;

**2/12/13**

Got Emily’s run file and some suggestions from her on what to try. Looks like she has a much smaller time step and also suggested that deepening the river channel where you force flow could be a thing to try. She also passed along a movie-maker code to look at output to assess where model is blowing up. Try this for above #5….

🡪 looking at depth averaged v, the highest velocities come in through the northern bndy still, at the corner there.

**2/14/13**

Make all depths in bay 5 m at start, then run again. Trying to make sure it’s the NE bndy condition and not something else. Call \*\_5m

🡪 went to NaN’s still! Need to confirm this came from NE corner…

next: do same run but halve the time step to 0.5 and use Emily’s parameters for ISplit = 5 (was 10) and Iramp =7200 (was 4400). Run on 12 processors.

🡪 blew up.

🡪 now try dt = 0.25 (half it again): this blew up again.

-> go back to 0.5 dt and use closure method for horiz. mixing, not constant.

**2/16/13**

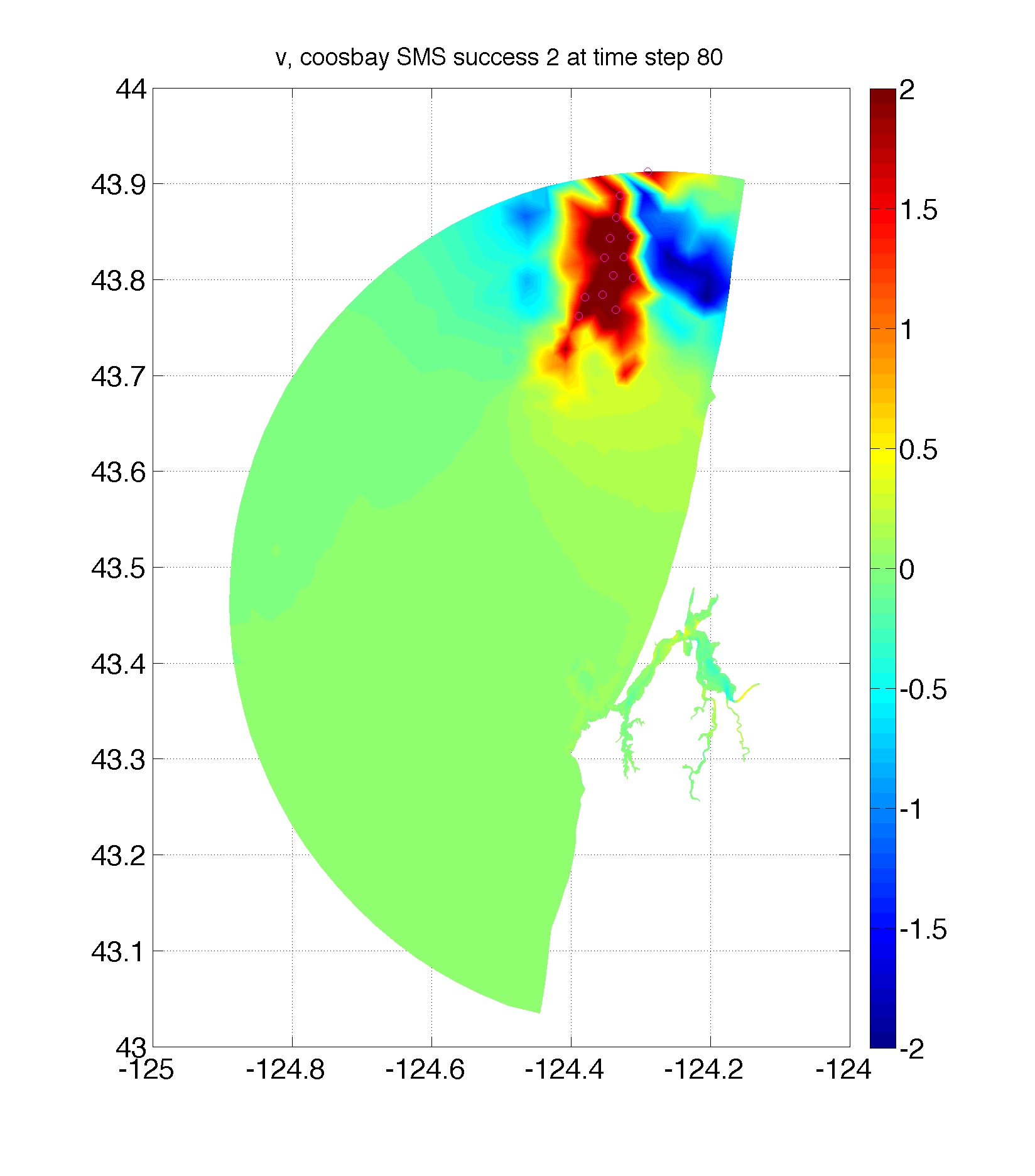
SUCCESS: Ran for 4 days without blowing up!

Changes: I switched horiz mixing type to “closure” but kept the horiz mixing kind to “constant” so I’m not sure it did anything, as I also changed horiz. mix coefficient to 0. Also changed bottom roughness lengthscale to 0.01 and bottom roughness min to 0.005. Not sure what really made difference, need to look at fields…saved in coosbaySMS\_success1.

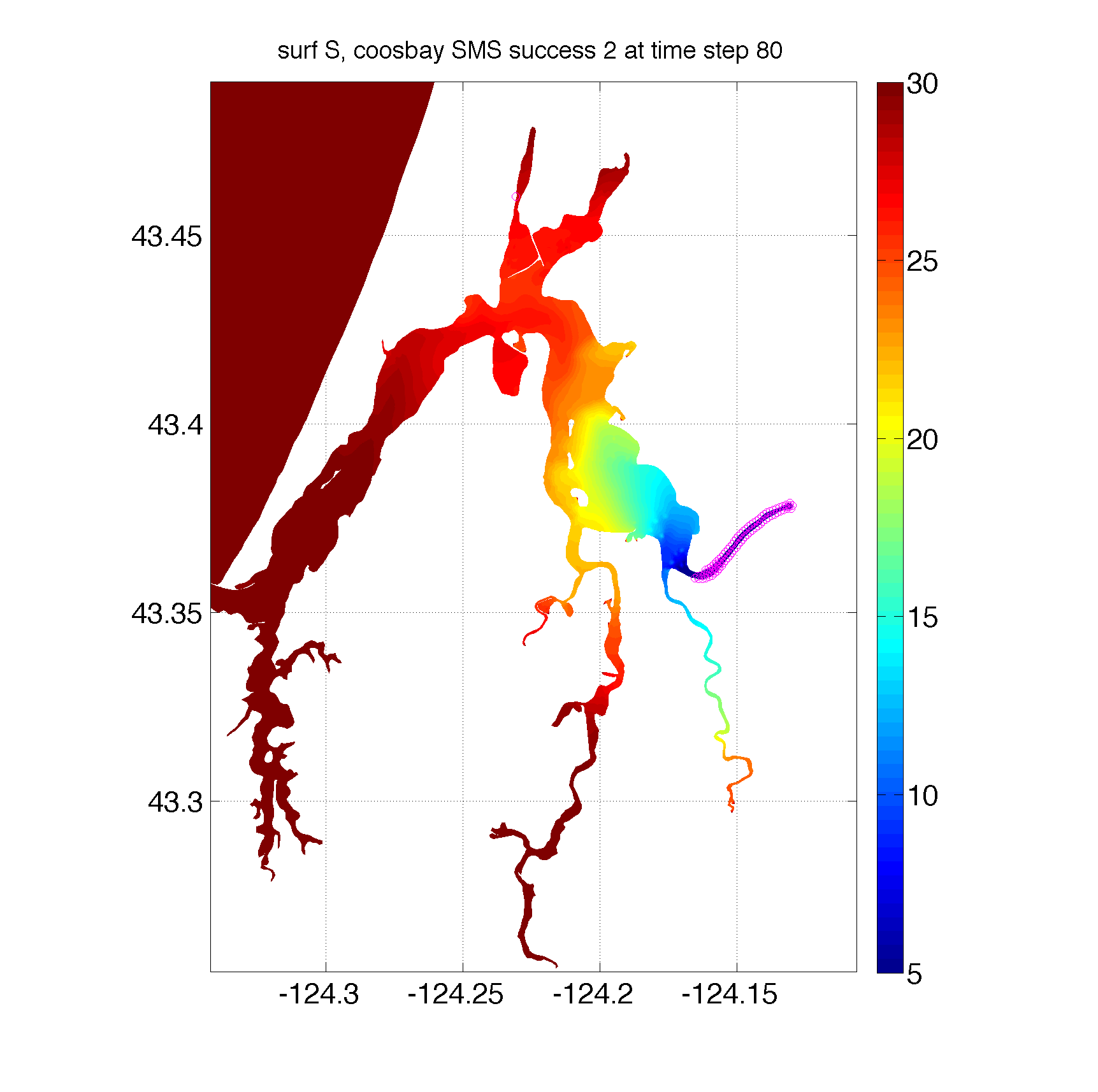
\*\*Still get crazy NE corner effects from bndy (high vel’s), but not >15 m/s, more like 5-6 m/s but still not good…need to diagnose this further, ask around. \*\*

Try getting rid of OBC nodes in that NE corner. Make OBC’s 1:48. Download to success2 folder and check out fields: yep same deal, though this time it doesn’t occur right at the NE bndy since I didn’t make those OBC nodes. So it doesn’t seem to matter where. Maybe I need a sponge layer on all these nodes too.

See figure on next page for example of what this looks like. Magenta dots are surface v > 3 m/s. (this example has tides, 1 river)



Surface v velocity for time step 80 hours into run for success 2 case. Still get instabilities forming at NE boundary though the model didn’t “blow up”.

This is surface S at same time step inside Coos Bay (everywhere else = 30 psu). Magenta dots are S < 1.

**2/18/13**

Add sponge layer to all OBC nodes (5 km radius, 1e-4 inverse time scale), not just river node. See if that helps control these instabilities….running on 14 processors. Check in later: So far, so good—almost done and the internal vel’s are small, while at same time step in previous runs they were edging up towards 10 m/s.

Need to verify and check: worked! at least for 4 days that I ran model, the instabilities don’t show up. Of course, I do have depths artificially deepened within estuary to 5 m, and only 1 river, but this is promising.

--🡪 saved in success3\_coosbaySMS folder in Run/

Next: This run I went back to original OBC’s (1:52, i.e. entire boundary) with sponge layer as before. Run for 14 days, using dt = 1 sec (instead of 0.5 sec as above). This should work, if not reduce dt back to 0.5.

NOTE: outputting only every 2 hours to save space, and only TS and horiz. vel’s.

Ran successfully for 14 days, outputting every 2 hours. 🡪 saved in success4\_coosbaySMS folder in Run/

Now try this with real depths (except modify sloughs and parts of bay that are too shallow to be deeper)—intermediate step: deepen Coos Bay to 2 m everywhere where it is less than that, and 5 m in Coos River still.

-> finished this intermediate run (2 m depth min) successfully too, with no nan’s. Doesn’t seem like any dry cells developed though, so next test is to use real bathy and see what happens…🡪 saved in success5\_coosbaySMS folder in Run/ for 10 day run using dt = 1 sec.

**Next**: Go back to original bathy, use -3 m as “min” depth, i.e. 3 m above MLLW. This doesn’t deepen any other sloughs than coos river, but see if it works anyway. Use same dt and run time as above. Running now…

SUCCESS! …🡪 saved in success6\_coosbaySMS folder in Run/ for 10 day run using dt = 1 sec. Haven’t looked at output; need to figure out to visualize wet/dry cells. Just wet = 1, dry ~= 1, so can nan out dry cells.

**2/19/13**

Started month long run: messed with NC\_OUTPUT\_STACK, which tells it how many time intervals to write to each NC file. 0 puts all in 1 file, 1 makes a new file for each time out. Also writing out restart files every 7 days.

🡪 success! Saved 1 file every 3 hours. Not sure what is better. Also wrote out 3 restart times to 1 single restart file. Saved in month\_coosbaySMS\_1/ folder in Run/

**3/2/13**

Met with Jim and Emily again—suggested looking back at Ralston’s Merrimack paper in JGR, where he uses fvcom. He used 20 vertical levels and a bottom roughness of 0.5 cm after tuning it. Also used 0 for horizontal diff and 10^-7 for vertical. Make these changes in the month long run and run again. In sigma.dat just change nlevels to 21, as we are using uniform sigma type that’s it. Also changed in .nml file, so didn’t actually have to remake any files to run this new case.

-> running on 10 cpu’s….

**Blew Up!** Bummer- after only 45 minutes model time, got huge negative free surface. Trying now with same set up but half dt, dt = 0.5 sec.

🡪 already blew up again. Go back and try changing one thing at a time…first use 10 sigma levels and change roughness length scale, keep vertical mix same as before.

\*\*\*oops- just realized I had put z0 to 0.5 meters not cm. Change this to z0 = 0.005 and see if it runs better…made it past where it blew up first time, so I that was one of the problems at least…now running on 14 cpu’s again, but only for 4 days model time.

…this ran successfully for 4 days!

Next: change dt back to 1 sec and run for 1 month, outputting only every 3 hours.

--still running, will take about 14 hours to run 1 month on 14 cpus…

**Blew UP!** But made it pretty far: blew up at

!400000 2008-01-24T03:33:20.000000

Not sure why yet—download output and look. Could restart with smaller time step (have a restart time step at 1-23-2008, number 4 in coosSMS\_restart\_0001.nc)?

**3/4/13**

Restarting—looking at Emily’s run file, change to ‘hotstart’ and specify file coosSMS\_restart\_0001.nc. Change types to ‘set values’ and comment out startup values. Change start date to 1/23/2008 which is last time step in restart file (not sure if this is right thing to do). Also change dt to 0.5 sec from 1.0 sec.

Try restarting: lots of errors: need to set start date to a time corresponding to a time in the restart file. Also, have to set first nc out to a date later than this start date, and probably rst too.

Got it started—note this overwrites a new restart file in output directory. What about history files? Yes! Starts from \*\_0001.nc again! So need, to specify a new output directory in future…

But running…finished, ta da!

Comments from Dave Ralston:

“if you have wetting and drying turned on, you can also improve the stability by increasing the minimum depth  -- small minimum depths and big friction make for an unstable model.  You can also test to see if it is a wetting and drying problem by using the depth offset factor in the \*nml file (static\_ssh\_adjust, I think) to increase the water level everywhere and see if it runs.  Also, restart files are your friend....”

**3/5/13**

Redo the month long run at dt=0.5 sec and with 20 layers, etc. Restarting caused all history files to be written over, so I am redo-ing it to be consistent. On 12 cpu’s 1 month will take about 1 day of real time…not the best!

**3/7/13**

The re-do worked and have hourly fields for a month at above parameters—however, the running at 0.5 dt took awhile, so it’d be nice to run it at 1 sec dt. Try to get this working by implementing some of Dave’s suggestions.

Justification—right now using MLLW, so adjust static\_ssh\_adjust by 1.30 meters everywhere as approximation (about that looking at VDatum from MLLW to MSL). Then change min\_depth to 0.1 instead of 0.05 m.

Try to run this at dt = 1 sec, saving every 3 hours only. Started on 8 cpu’s….

**8/14/13**

Downloaded latest release 3.1.6. Uploaded to ocean machine and am now trying to ‘make’ by copying most of the make.inc file from the old source folder to the new one. This means using lots of the same target directories as before (point to intel compiler, etc.).

--it compiled. Now test with running CoosSMS test run for 2 weeks.

Now running—had to change nc\_out\_interval to 3600. and nc\_output\_stack=0 (did both at same time) and first file out to 1:00 hour in, or else I got weird fatal error about floating point numbers for the time variable. But it seems to be running now…

🡪 however, it might break again at 1 hour when it tries to write out the first time his in the output file.

Yep. Still broken. I looked at the netcdf output file and everything is there, but the time variable is size 0. Probably what’s messing it up.

**11/8/13**

Things to add to the model:

1) Realistic tidal BC’s

2) Realistic river forcing (and add more rivers)

3) Realistic IC’s from our data

4) Realistic BC forcing for T/S from Parker’s model

5) Realistic wind/met forcing from atm model used in PNWTOX

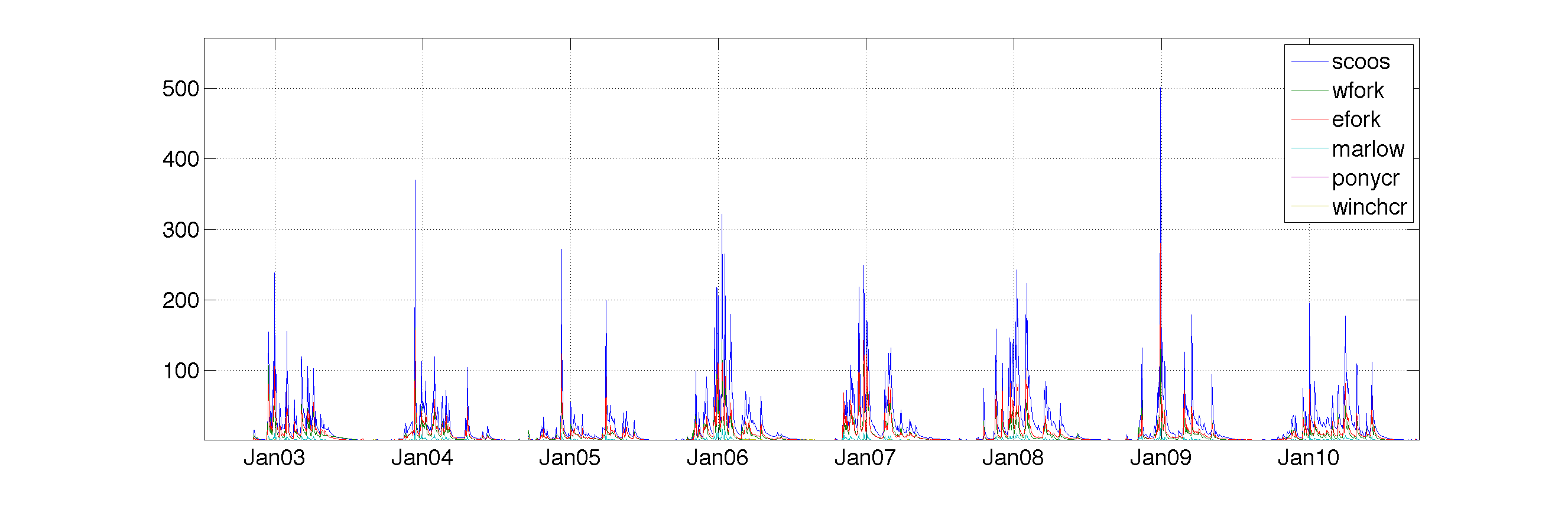
I have the BC files from PNWTOX for outside Coos Bay for the years 2004-2007. I have atmospheric data for 2001-2008, and I have processed these into mat files for the Coos Bay domain for 2004-2007. All this in BCs\_Parker folder in HyCoo directory.

I have river data for several rivers from the CWA for nominally 2003-2010 (for S. Fork Coos River, for example). Associated temperature data from rivers too, but only for limited time and smaller rivers—but it’s better than nothing. All this in Coos\_Water\_Assoc folder.

Need to decide on good time frame to run model—when to initialize and let it go. Have good hydrography covering most months, but in years with forcing, i.e. have from Nov. 2012 – Nov. 2013. Interpolate T/S in coos bay grid to nearest CTD location, and then fill down to bottom and to surface—this is similar to what I did in Puget Sound, so look back at that code. In Puget Sound I wrote out an initial file using global model, then overwrote that with Z\_PS\_iniCruise.m, then used Z\_PS\_ini\_rivers.m to make S go to 0 at river input points.

Try one thing at a time. First, try to put in variable river forcing. Let’s go for 2006-7: need to write river forcing file in time format that fvcom recognizes: thisis modified Julian days. Use datenum2mjd.m to convert Matlab datenum to MJD, and use mjd2nu.mm to convert back (in jlab).

To make 1 river file for Coos River from CWA data—add together SF Coos, EF Millicoma, WF Millicoma, and Marlow Creek. Not all data complete in all years—but have relationships between each river based on times when have them all—i.e., can fill in gaps based on other rivers. Use pfit’s in discharges\_CWA.mat file to do this. See make\_CWA\_rivers.m. Now have interpolated Q,Temp for all days Oct. 2002-Sept. 2010 in the file Q\_TEMP\_interp\_CWA.mat for those 6 rivers. (see figure below in m3/s).



IDEA: to run starting Sept. 1, so that it runs for a month before being hit by big discharge, for example Sept. 1 2007, there’s not a big discharge event until Oct.; this would give it time to equilibrate—need to initialize with CTD data that reflects this low flow state (remember Sept. 2013 had a couple big rainstorms, so make sure it’s before that).

**11/9/13**

Check out FVCOM 3.1.6 user manual and all matlab code that came with it—lots there for writing netcdf forcing files and for plotting fvcom results.

**11/11/13**

Uploaded files for HyCoos test #1, which is the same as above but has realistic river forcing for the Coos River. It also has temperature data for the river time series. I changed the start date to 9-15-2007 and will run for a month to go from low discharge to high discharge events.

Note, also using old 3.0 version of FVCOM, until I get 3.1.6 to run. I emailed Chen about this.

This run is called hycoostest on server and the files are HyCoos\_TEST\_\*. Need to wait and see how many processors left, as Dustin is running…then run

mpirun -np ## ./fvcom --casename=HyCoos\_TEST --logfile=log &

Now running on 8 proc. (stopped, see below)

NOTE:

1) Have to change all filenames and river names (in river .nml file) when you start renaming things.

2) Check STATIC\_SSH\_ADJUST: It seems like since negative is above sea level, that making this = 1.3 is actually making my bathymetry shallower by 1.3 m everywhere, as my MIN Depth is -1.7 m (when it should be -3 m). Next time switch this to -1.3 m (or 0) and see what happens.

Restart—motivation was to restart and do a month that sees more discharge, as I just looked at Sept 15-Oct. 15 2007 and it doesn’t do much. Try to simulate month of Nov. 2007 (Sept. 15 might be a good start date for a future year long run, but not for now), which has a large discharge event, after starting small.

Also, make change to SSH ADJUST, to 0, and output only every 3 hours. Now running.

This makes mid depth -3 m, as I thought. However, maybe I do want to ‘deepen’ everything, which is what I was doing before. So restart with SSH adjust to 1.3, now running again starting on Nov. 1, 2007.

**11/12/13**

HyCoos\_TEST still running, now on Nov. 26—I am stopping it here and seeing if the output is reasonable. Save to hycoostest folder in the Run directory on my laptop. Got to 202 output files (every 3 hours). Looks reasonable based on log output (i.e. all variables were still non nan’s).

Also, got the following email from Jianhua Qi with FVCOM about fixing the 3.1.6 error above:

“Can you try setting DATE\_REFERENCE in namelist NML\_CASE as 'default' or '2008-01-01 00:00:00'?“

So, next I will try the same case as above, but with 3.1.6.

So, I added DATE\_REFERENCE = ‘default’ and it is running, but we’ll see if it writes output. In the original run\_nml file DATE\_REFERENCE was not set to anything. Seems to be writing netcdf output files fine now. Download one and see. YES, working. So stop this run and switch over to FVCOM 3.1.6 for HyCoos tests in future.

HyCoos\_TEST looks good—get big discharge event that pushes salt out. Now need to add other rivers and IC’s, etc., etc.

**12/5/13**

Emily (OSU) still having problem with sponge layer near OBC—looks like the 3.1.6 version she had didn’t have the fix in it, but mine does…so good there.

Looked into using HyCOM or NCOM for OBC’s. This would allow me to do more current years. Emails from Lucy Smedstad have info and links to these model output fields—could be fine. Doing 2012 would allow us to use data we’ve already collected. Just need 2012 CWA discharges.

**4/4/14**

Got 2011-2013 discharges from CWA. Put into matlab format and correlate where needed to get discharges. Note that in talks with Jon Souder and others, there is some FW input at the ends of each slough, such as Isthmus slough, etc. First approximation would be to scale these in watershed area and put in some discharge—they are all tidal though, so there are no good estimates of discharge, plus Jon thinks this scaling won’t be a good approx. b/c the watershed characteristics are so different. But, it’s a place to start.

This file is saved as Q\_TEMP\_interp\_CWA.mat in CWA folder in Coos\_Bay directory. Next, I need to find info on watershed areas for Isthmus Slough, Catching Slough, etc. and come up with discharge estimates for those, then implement this in a test version of HyCoos (most likely will need to deepen these sloughs artificially until have better bathymetry, see below).

In mid-March, Jeff Wood and Peter Ruggiero took bathy data from jet-ski’s in the bay. Need to get this data from this and work it into the bathymetry for the model.

Edited write\_FVCOM\_river\_das.m to handle multiple rivers. Now need to edit make\_HyCoos.m to deepen the sloughs/rivers of interest (only do this for Coos River right now), then add in river nodes where I want them to. Add this info to bathy\_polygons\_CoosBay.mat file and work in to make\_HyCoos model maker.

Right now have:

1) Coos River that combines South Fork Coos, West Fork Millicoma, East Fork Millicoma and Marlow Creek; Tioga Creek is gaged, but the SF Coos gage is below this; however, since I’m missing some of the watershed area for the Coos, just add in Tioga for now

2) Winchester Creek: at south end of South Slough

3) Pony Creek: empties into north bend of estuary

NEED: inputs into Haynes Inlet in north, Isthmus Slough, Catching Slough, Coal Bank Slough, Joe Ney Creek, Elliott Creek;

Updated bathy polygons file to have:

1) Polygons around areas in Coos Bay to interpolate CTD data to for IC (box)

2) River polygons to deepen, find discharge node, and make thalweg line to take IC to S=0 along (rivbox). There are 7 of these right now: 1 for Coos, 1 for South Slough, 2 for Haynes Inlet, 1 for Isthmus, 1 for Coal Bank, 1 for Catching

3) OBC box to make vertical wall along northern coast as I was doing before;

In terms of building an IC, look back at Puget Sound code on how I did that with prism cruise data. Want to do similar thing but with our data—could start with Nov. 3, 2012 CTD survey, which is the best and earliest transect we have, i.e. run for month of Nov onward through when we have discharge data (to 9/30/2013). Then once have updated discharge through 2013, can run longer. Also need IC for shelf and BC’s for shelf. Maybe look back at HYCOM and what Lucy said…

**6/13/14**

Worked on coastlines some—got new digital coastline from noaa chart, need to combine this with the old coastline I have which is better in some places. The old coastline is used by plot\_WAcoast(‘coosbay’), while the new coastline is in FVCOM folder in GE\_cusp\_NOAA\_shoreline/OR0501 folder. Then add in bathy and make new grid using SMS.

The bathy file is coosbay\_int.mat, and is in the folder /Users/daves/Documents/Coos\_Bay/Bathy\_data/mbsystem\_combo/data

New run: try starting HyCoos Test on 10/1/2012 and run for 2 months through Nov. 2012; In addition, I am going to add in the new bathymetry (but on the old grid) and a constant but time variable wind (from Port Orford buoy met station—see data file for how this is correlated to South Slough met station).

**6/14/14**

Made wind forcing file from Port Orford buoy data using write\_FVCOM\_wind\_ts\_speed\_das.m, which could easily handle spatially variable wind in the future. Set wind on in run file, and put type to speed and kind to variable. Also using new bathy and running starting on 10/1/2012, using FVCOM 3.1.6 (add DATE\_REFERENCE=default as above). See if this works…use 16 processors.

2 things: 1) something weird with time variable in wind forcing file, wants it to be unlimited. if I turn this off, it gets past this, but

2) elevation file doesn’t have time series until 2012, so need to add this in…

Went back to matlab file to write wind and made time unlimited for the wind file. Also, updated tide forcing file for Port Orford to go until 2014. Upload these and try this again…

🡪 need ‘source’ global attribute in forcing file. Add this…

and now it’s **running**! Will check in on it later to see how it’s going. Have 10 cpu’s. And saving every 3 hours.

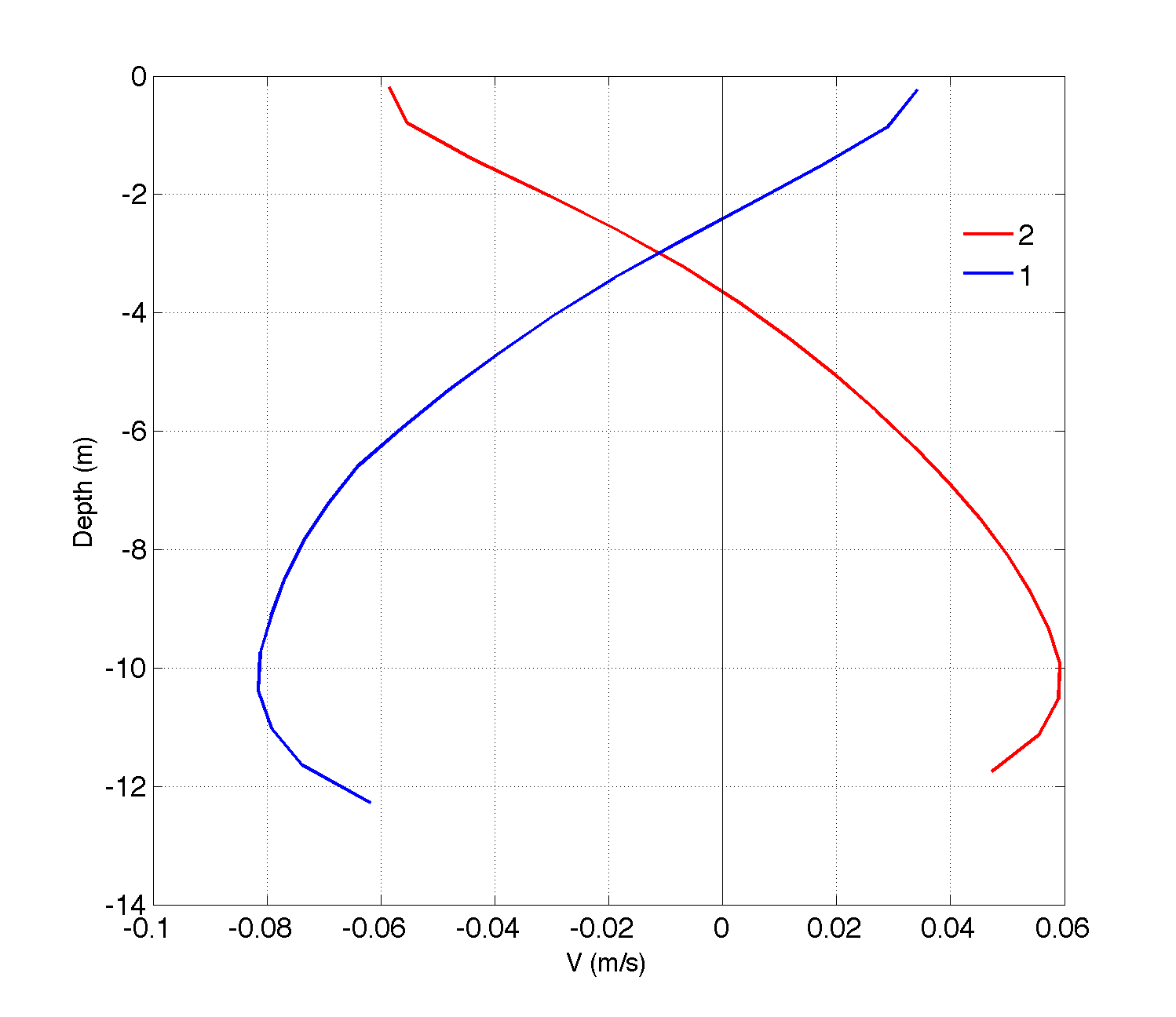
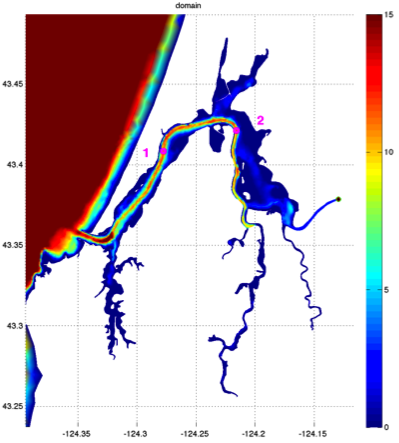
**6/19/14**

Latest hycoostest run finished! 2 months of model time in about 1.5 days, every 3 hours saved is about 4 GB. This run has different bathy and winds, and is running from Oct 1-Nov. 30 2012. Need to download all output files to laptop.

6/20/14

Got all files downloaded. Made surface S movie—looks like everything was working, but the estuary gets too fresh during the big November discharge event. Maybe this would be fixed by using the new bathy/coastline file where the Coos River source is put farther away, and is deepened.

Also—used the mooring extractor code to get a couple time series of S and velocity profiles at locations in the channel (see plot below). The salinity profiles show some slight stratification, but the mean v profiles are opposite to what I expect—and I can’t figure out why! Mean flow northward in surface at location 1 and southward at surface at location 2 (see plots). It’d make sense if the profiles were flipped upside down, but I’ve checked a lot of times about how to make the depth variable and its correct. The same thing was occurring in the previous model runs, I just didn’t document it. This doesn’t make sense because in the surface S movie, you can see the freshwater moving out. Something fishy…



plots for hycoostest\_2, domain on left and 2 mean v profiles on right.

Basically, it looks like if I plotted **–u and –v everything would look right**!

On the other hand, I looked at water levels from the Charleston tide gauge vs. that measured near a spot close to Charleston in the model. The willmott score was 0.99 and the skill score was 0.98 (see Ralston et al 2010)—there are some issues, but some of it is due to using 3 hourly output. Some lows are overestimated it looks like, but good impression overall! At least that works.

**6/23/14**

Got some replies from Ralston and Geoff Cowles. Something’s up with the node numbering. It is supposed to be clockwise around an element face in FVCOM, but for some reason, mine is not coming out that way. This might have to do with the SMS mesh I am using, where in make\_HyCoos.m I switch the x/y to get it right, but don’t switch anything in the tri variable. Try switching this and see what happens.

Action—switched 2nd and 3rd columns in Mobj.tri (in make\_HyCoos.m) and wrote grd file from that. Uploaded this to server and am running same run but with these node numbers switched. Running only 1 month (Nov. 2012) on 10 processors.

WORKED! Just looked at 3 days of surface velocity and the tidal currents go in the right direction now. Whew. So need to make sure the tri variable in the grid is right if I ever make a new grid with SMS.

Redo hycoostest2 with this grid…so have all right variables. Only other change was to make IC for S = 32. Starting run on 10/10/2012 and go through 11/30/2012. Called hycoostest\_3\_triswitch on local machine.

**10/24/14**

Goal—make a new run for a low discharge case, using all same parameters as above. Need to fine a suitable time period that a) has low discharge and b) covers time period we have observations.

Maybe use April 2013 through Jul 12 2013 (gets a few CTD transects and starts off with some discharge in order to spin things up, then discharge falls off).

Change initial S = 33 to match Apr obs. No temp or anything else different, other than running for a longer time.

Use mpirun -np 16 ./fvcom --casename=HyCoos\_TEST --logfile=log & to run, after using ln -s /home/davesutherland/FVCOM/FVCOM3.1.6/FVCOM\_source/fvcom fvcom to make new executable.

Called lowQ\_hycoostest on ocean server. Started on 16 processors today, running! Finished and made some preliminary comparisons against obs.

**4/24/15**

Going to do the high Q run with temperature active and save hourly. Also saving turbulence fields out. The idea is to have some runs ready to test particle tracking code that Leif and others have been working on.

Called highQ\_withTemp. Modified the nml file to turn temperature active (and set it to T = 12 degrees as IC). Also, save every hour now (nc\_out\_interval = ‘seconds=3600.’) and save turbulence parameters (nc\_turbulence=T) and vertical velocity. There should be temperature variability introduced by the river—it has a temperature time series associated with it. Otherwise, there is no other source, as no surface heating or oceanic waters are forced.

Other things to work on: need much more resolution in channels, and make sloughs deeper. Also work on making initial conditions that mimic along estuary gradient and have constant forcing on ocean boundary (or climatology). Finally, need surface fluxes from meteorology reanalysis.

…Running on 16 nodes now: mpirun -np 16 ./fvcom --casename=HyCoos\_TEST --logfile=log &, in highQ\_withTemp directory.

The history files are now 18 MB, increased from 8.7 MB when saving T, W, and turbulence parameters. Saving every hour increases size needed for storage too. For 2 months, this is 30 days \* 2 months \*24 hours\*18 MB = ~26 GB (1440 files).

**9/14/15**

Working on matlab code to write IC file from CTD observations. To start FVCOM from observed T/S values, you write to STARTUP\_FILE = ‘CBAY\_ini.nc’ or other named file that has arrays of T/S in it, and time. I need to figure out the time variable. I assume it looks for the time that you start the model in the \*.nml file. The Sea Ice example in 3.1.6 folder has an example initfile.nc that it reads from, although that case seems to use fixed z-levels instead of sigma levels. Also, there is a write\_FVCOM\_obs\_TS.m file in the pre-processing directory.

Here, the goal is to initiate the model based on CTD obs from a certain month, using mouth observations to do the whole outside area, and filling in the sloughs and other areas missed by the CTD in a smart way. Based on Puget Sound work. Make S go to 0 at head of ‘rivers’, which need to be identified.

Matlab file is in Model Maker folder: Z\_CBAY\_ini\_polys.nc, as it uses defined polygons to help smartly fill in IC.

So far—today I made it work with one polygon (outside CoosBay) and to write constant values to ssl and tsl. I am going to see if this works now in the run folder highQ\_withTemp\_IC.

Tried to run—crashed. Need ‘zsl’ in startup file. And SUCCESS! That is, you give it standard depths (zsl) with z upwards in the ini file and FVCOM must interpolate it itself? Need to test with non-constant IC’s. But for now, adding zsl made it start running!

**9/15/15**

Kept working on IC code. Now it works in bringing in CTD data and interpolating to standard depths. I also added 3 more polygons: south slough and 2 in the main estuary. These work okay, but I’ll need to add river/slough specific ones too, where I interpolate from the main channel to the river/slough end, going from observed S to 0 at the end. Keep temperature constant at the observed.

For now, try running with the interpolated observed T/S IC’s: Working! Download first few run history files and investigate them. Seems to work! yippee.

**9/16/15**

Now working on IC rivers code that will make river ends go to 0 psu and the right temperature for the time of the initial condition. Although I’m still only forcing one river (Coos River), make river polygons for all the sloughs/inlets for future. This code is Z\_CBAY\_ini\_rivs.m. You must run the Z\_CBAY\_ini\_polys.m first (or have another IC file) before doing the river IC’s.

Have 10 river polygons. The code works in matlab, now try the IC in FVCOM. Run on 24 nodes the highQ\_withTemp\_IC case. It’s running—let it go for awhile and look at first and last history files.

Result: Worked, ran fine with no hiccups for 4.5 days. Looking at the last history file, the low salinities I imposed in the sloughs/inlets still there but starting to get diminished by tidal exchange of salty water, since no discharge into those areas, except for Coos River. Could work on adding these river systems next?

**Adding rivers**.

Try adding another river input as a node input into the South Slough. Use Winchester Creek discharge x 2 (don’t worry about correcting for watershed area yet) using rivbox variable in bathy\_polygons\_CoosBay.mat file and the code already in make\_HyCoos.m. That is, make nRivers = 2.

Take approach to write netcdf forcing file for each river. Thus, change RIVER\_NUMBER to 2 in run nml file, but keep same river info file. In that river info file, add another section for the new river. That is add the RIVER\_NAME, etc. and point to the new netcdf file, and specify the node number (7875) for this particular model grid. These are in the run folder HighQ\_withT\_IC\_2rivs.

Then rerun make\_HyCoos.m to get new input files, including new obc.dat and the river netcdf files. Finally did this, lots of bugs to fix. See if it works now, upload to ocean and run.

Result: Running! Had a couple more bugs to work out. Note names in netcdf file have to match the names in the namelist file for the rivers. It seems like you could easily write multiple rivers to one netcdf file actually….deal with this later. For now, let it run, output history files every 3 hours and look in a few days…stopped run, as it was working okay.

**NOTE**: The Winchester Creek flux is really small, even when doubled. It’s like 1 m3/s even for the Fall 2012 ‘highQ’ case. Maybe need to up this to see effect…no, you can see that it’s outputting FW in South Slough, even when it’s this small. Need to look at watershed areas, and scale accordingly. Then add in other rivers elsewhere.

To do: add in more rivers, scale flux accordingly. Make it adaptable to handle a new grid/mesh in the future.

**9/17/15**

Added rivers, so that now there are 12 fw inputs into the model. To do this, I edited bathy\_polygons\_CoosBay.mat to add node positions to the rivbox variable. This way the code picks out the node and outputs in make\_HyCoos. Then it writes a separate netcdf file for each river. Note that the names of the river in the rivbox\_names variable now is the same name I use in the namelist (and the nodes need to be specified there too).

Right now this 12 river setup is running in the same folder as the 2 river test case above. I specified fluxes based on watershed area to some degree, although this will need some refinement. Right now, I want to see if it runs.

Finished! Yay. Looked at surface S and T fields and it looks really fresh, i.e., maybe putting in to much FW either in the IC or in the river input. Jon Souder told me that most of these tributaries are actually gated: “Inflows from the tide gated tributaries to Coos Bay (North Slough, Palouse, Larson, Kentuck, Willanch, Catching [Stock and Ross], Coalbank, and Isthmus [Noble]) will make things more difficult because there will be inflows only on ebb tides, and a reservoir pool is built up behind the gates during flood tides. So they'll be a dampening effect, and then a pulse. “.

**9/18/15**

Try a run starting with constant S and T and run out to equilibrium, comparing to run with IC. Call this highQ\_withT\_rivers (no IC). Only change have to make is specify constant in run file and set values. Set S = 30, and T = 12.5. And run with same 12 rivers.

Running…finished.

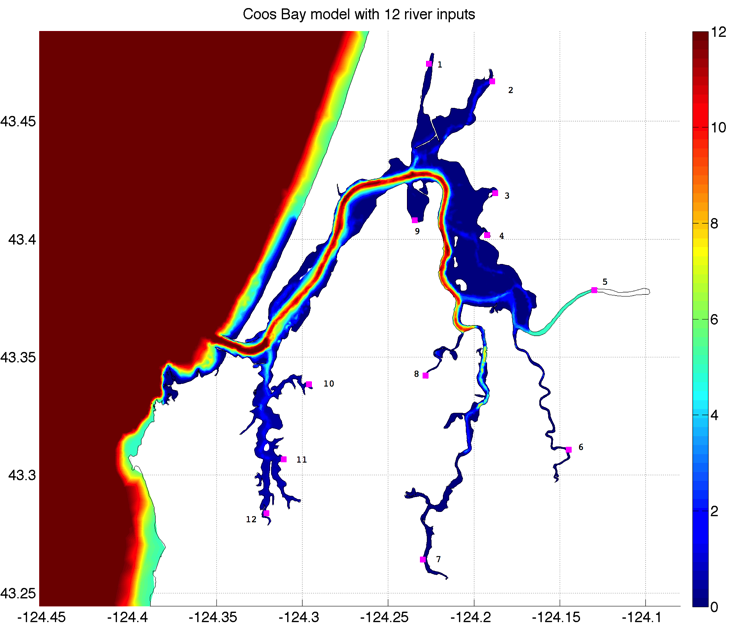
**9/21/15**

Looked at surface fields of constant IC run with rivers vs one with IC built from CTD profiles. Look very similar after about 2 weeks….movies saved in run folders, as well as mooring extracts for same positions.

In both cases, looks too fresh to what we’ve seen in CTD transects. Maybe adding too much river discharge. But, that’s easily fixed (or at least varied).

**9/30/15**

Got river watershed areas from CWA (Jon Souder and Freelin Reasor), now in xls file in CWA folder on computer. Putting in FW at following locations (numbered map). He suggested adding a 2.5 (in between 2=Palouse and 3=Kentuck), which is Larson Creek.



Also, the scaling using his watershed areas and using Winchester Creek time series, make FW discharge much lower overall. Try this again with 12 rivers, using new discharge. No IC (constant IC’s). Running now as highQ\_withT\_rivers/. Saved as try2 on my laptop.

Done running—still seems like too much FW. In reality, changing those small rivers doesn’t change too much the overall amount of FW. The big input is still the Coos River. Maybe need to add depth and channel to Coos River and to all other systems, so there is more estuary. Probably also need to increase resolution.

**10/2/15**

Checked discharge values again—they are correct. So, to get more salt water, need more wind mixing, deeper depths and distributed profile of discharge (i.e. coos river channel might need to be deeper).

Run again and output wind velocities to just check they are doing what they’re supposed to, and change bottom roughness to 0.01 instead of 0.001 (and change minimum in nml file correspondingly). This is more in line with Emily Lemagie’s Yaquina runs. They also used vertical mixing coefficient of 1e-7 when I had 10e-7, so I changed that too.

Run in the same directory on ocean, then save to try3 on my laptop. Wind looks like it’s working fine, getting time-varying, spatially constant values that I input from Port Orford buoy.

**Low-S problem….**

Doesn’t look like this is going to change anything dramatically. Looking back at Nate Hyde’s thesis, they ran into the same problem, which is why they added ‘deep’ mixing pools at each fw input (except south slough). This helped bring more salt into the estuary and mix the FW in. They tried extending the river channels but said that had a negligible effect. Need to see how long of a channel I am missing….

My eastern most point for Coos River is near -122.13, from there it’s about 16 km to Dellwood on South Fork Coos (about 2.3 km to the fork of South Fork and Millicoma).

From confluence of SF Coos and Millicoma, it’s 14 km to where EF and WF Millicoma Rivers come together at Allegany. Head of tide is just up each of those forks from there. So, really, have mixing occurring on Millicoma about 17-18 km up from where I end my model currently, and on the SF Coos it’s 1 km up from Dellwood, so about 19 km missing. Or split forks into 2 separate rivers. Need to try this next—but that means messing with model grid. Will also update bathy at same time? Ask Kai at OSU for latest.

**11/3/2015**

Got new grid/bathy from Kai, based on the USACE Lidar bathy I gave him, plus others. It’s on an ADCIRC grid, so I will use it to interpolate to my original grid. I will also try and extend my Coos River channel out another 19 km. So, change 1 thing at a time:

1) update bathy using Kai’s new one, but same coast boundary:

Running under highQ\_newZ folder.

Note that I had to add a “&” to my mpirun command on ocean, for some reason.   
So, now run command is: mpirun -np 16 ./fvcom --casename=HyCoos\_TEST --logfile=log &

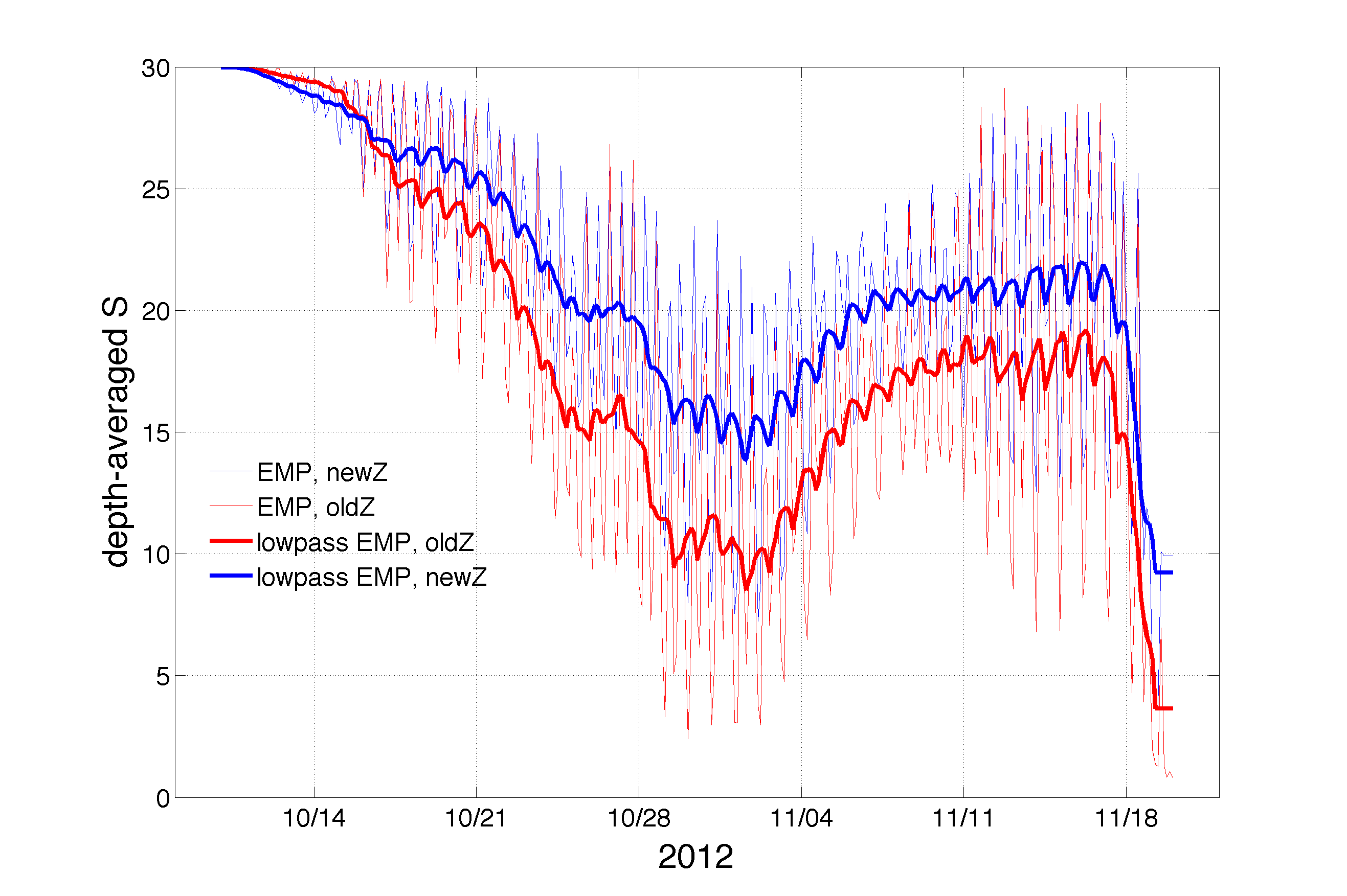
**11/4/2015**

highQ\_newZ run finished, although it got NaN’s the last model day. It looks like the velocities started getting really high for some reason—most likely bathymetry not being deep enough in some of the channels where I am putting rivers in.

But, overall, the new bathymetry makes the volume in Coos Bay larger, as calculated with Ralston’s getGridVar.m code that gets areas of the elements. Also, the average S in the log is higher at similar times compared to the last run with the old bathy (and same everything else).

So, need to work on grid and bathy some more, extend river channels (deepen some).

But, the depth-mean S at Empire was raised 5 psu in the new model run…see below:



However, compared to data at EMP and BLM, the model is still too fresh—in that it freshens too much and too rapidly. There is an offset b/c of the IC, but this wouldn’t help the timing and rate of change in S.