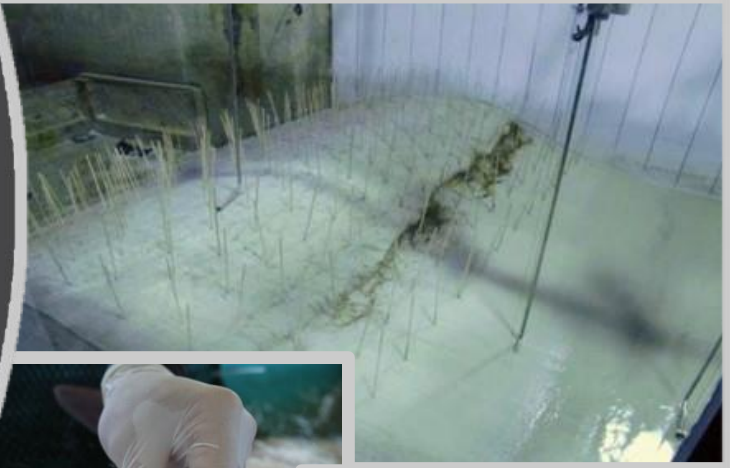
CIRP Technical Discussion

Update on the development of CORSED: Consolidated Sediment Transport Code

Earl Hayter and Joseph Gailani





US Army Corps of Engineers

ERDC ENGINEER RESEARCH & DEVELOPMENT CENTER

CORSED: Consolidated Sediment Transport Code

Project Delivery Team (PDT)

Gary Brown, Model tester

Mitchell Brown, CHL, Model tester, GIT

Yan Ding, CHL, Model tester

Joe Gailani, CHL, Co-PI

Terry Gerald, EL, Library Developer

Earl Hayter, EL, PI, Model & Library tester

Sung-Chan Kim, CHL, Model tester

Collaboration with Other Organizations

- RSM
- DOER

CORSED: Consolidated Sediment Transport Code

There is not one “best-practice” for sediment transport modeling. The methods selected for appropriate simulation of key sediment transport processes is heavily dependent on site conditions and is part of expert best-professional judgment

Background: ERDC maintains multiple sediment transport codes, each with differing objectives, advantages and limitations

- SEDLIB
- SEDZLJ
- CMS-SED
- Legacy codes, such as TABS

Purpose: Develop a single library sediment transport framework

- Include all features of existing codes
- Flexibility to permit adaptation to site-specific requirements
- Independent of hydrodynamic model
- Permit new R&D using a single platform

OBJECTIVES

- Develop SEDZLJ-Lib-v1, link it to GSMB, and thoroughly test both SEDZLJ-Lib-v1 and the linkage to GSMB.
- Develop and test the initial CORSED sediment transport library structure shown in Figure 1a.
- Link SEDZLJ-Lib-v1 to CMS and to ADH.
- Initiate the development of the second CORSED sediment transport library structure shown in Figure 1b.

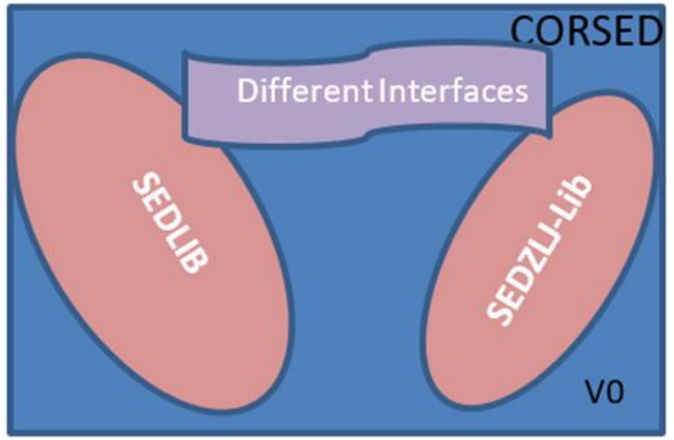


Figure 1a – Initial CORSED Structure

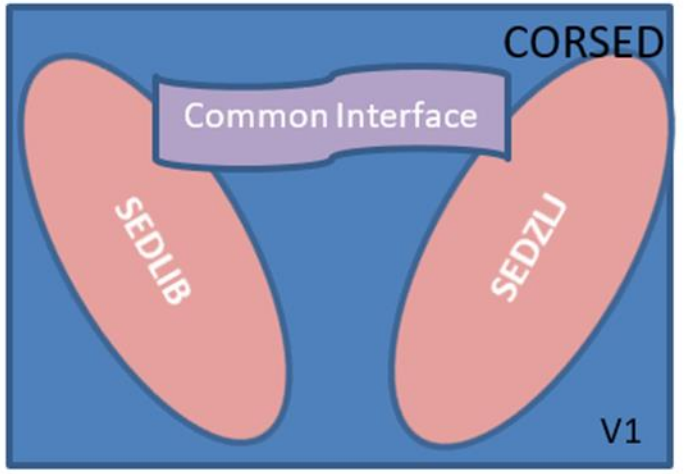


Figure 1b – Second CORSED Structure

CORSED: Consolidated Sediment Transport Code

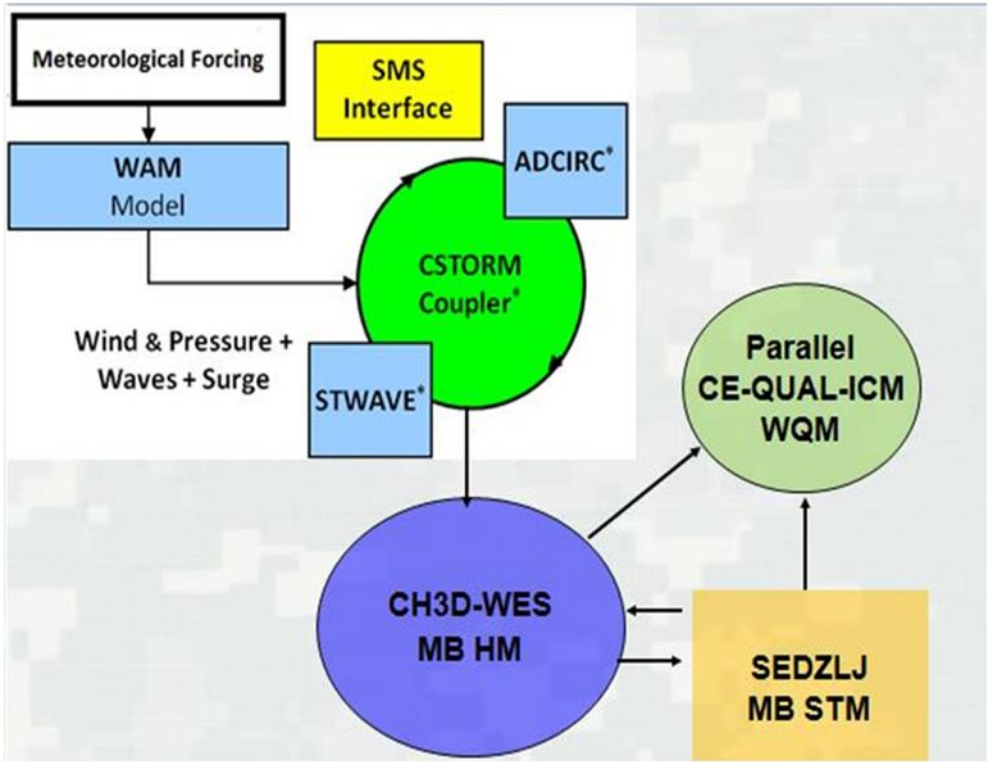
Scope of Work

1. Develop and thoroughly test SEDZLJ-Lib-v1, link it to GSMB, and thoroughly test both SEDZLJ-Lib-v1 and the linkage to GSMB.

    a. Create a GIT repository for the existing GSMB code and its hardwired version of SEDZLJ.
    b. Within the GSMB code, rewrite source code to isolate all SEDZLJ code pieces into callable subroutines. Separate out the callable subroutines into a SEDZLJ module that resembles as closely as possible the structure of the SEDZLJ-Lib-v1 modules. The purpose of this is to make replacement of the old SEDZLJ code more straight forward in the next step (c).
    c. Continue restructuring the CH3D-MB code to create a linker code between CH3D-MB and the SEDZLJ-Lib-v1 library. In the process of creating the linker codes, the isolated callable SEDZLJ routines from (a) will be replaced with similar calls to SEDZLJ-Lib-v1.
    d. Perform model validation on two existing GSMB model cases to ensure that GSMB with SEDZLJ-Lib-v1 gives the same results as the existing GSMB model, or that any differences are understood and acceptable.
    e. Update GSMB documentation to reflect the version that contains SEDZLJ-Lib-v1.

Task 1d – Model Validation

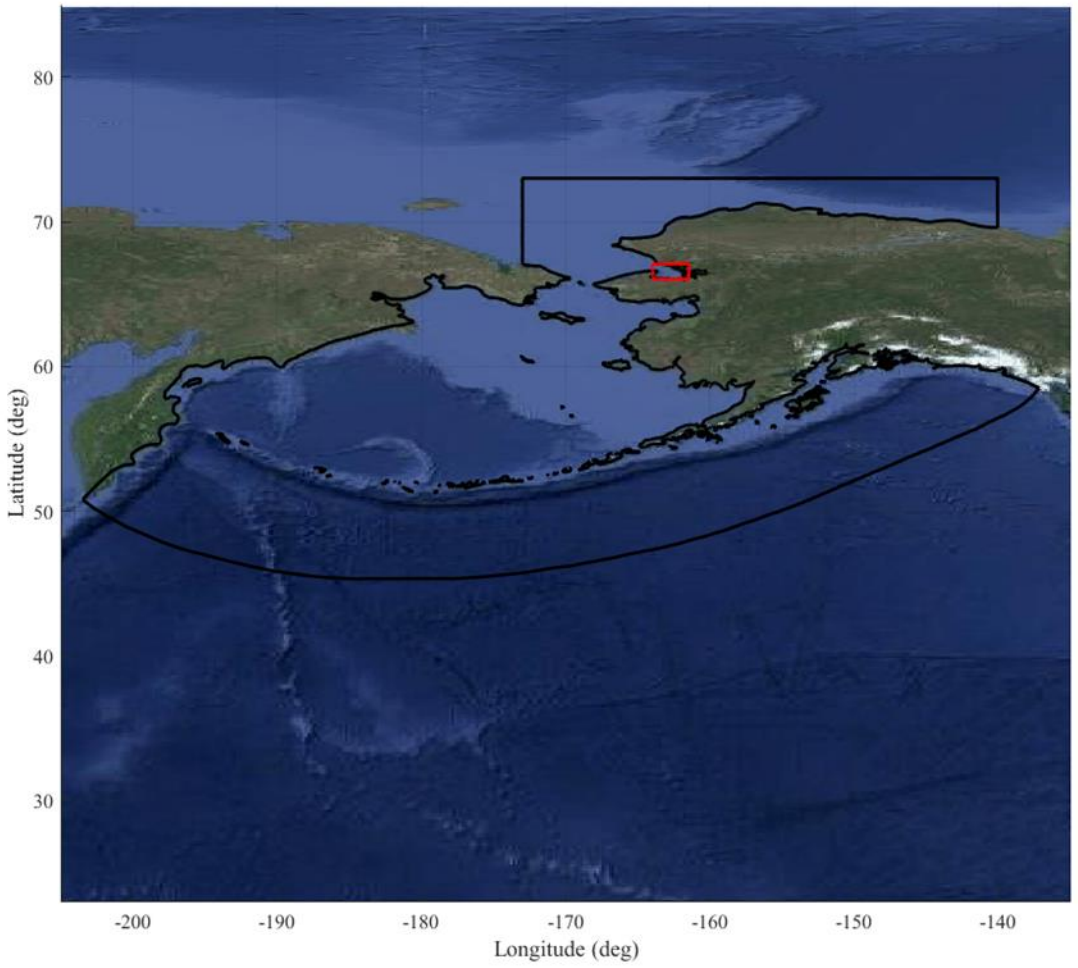
Perform model validation on two existing GSMB model cases to ensure that GSMB with SedZLJ-Lib-v1 gives the same results as the existing GSMB model, or that any differences are understood and acceptable.



Multi-Block Geophysical Scale Hydrodynamic and Sediment Transport Modeling System (GSMB)

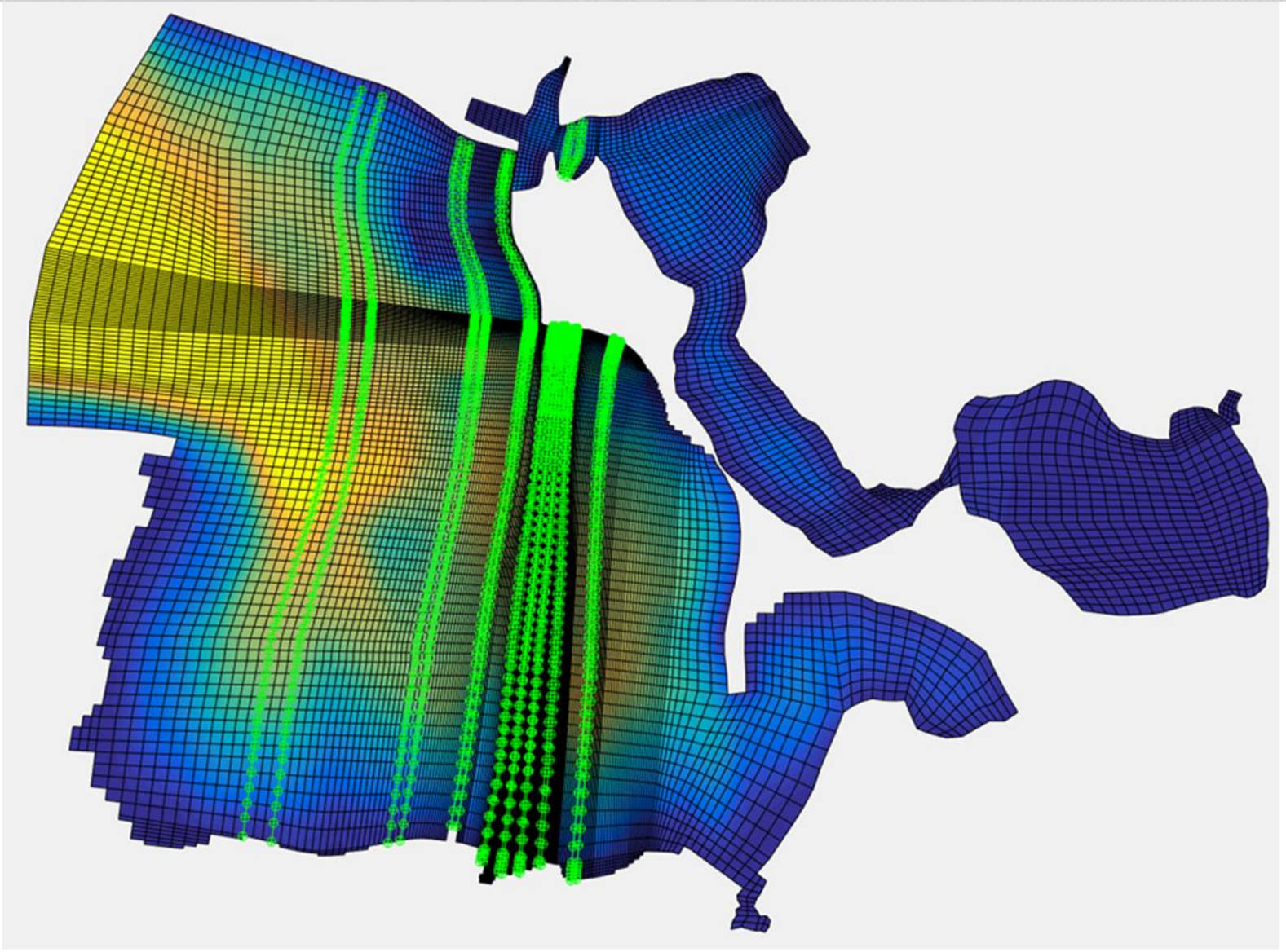
Task 1d – Model Validation

2D Test Case: Development and Application of a Geophysical Scale Hydrodynamic and Sediment Transport Modeling System: Kotzebue-Blossom Point Navigation Channel

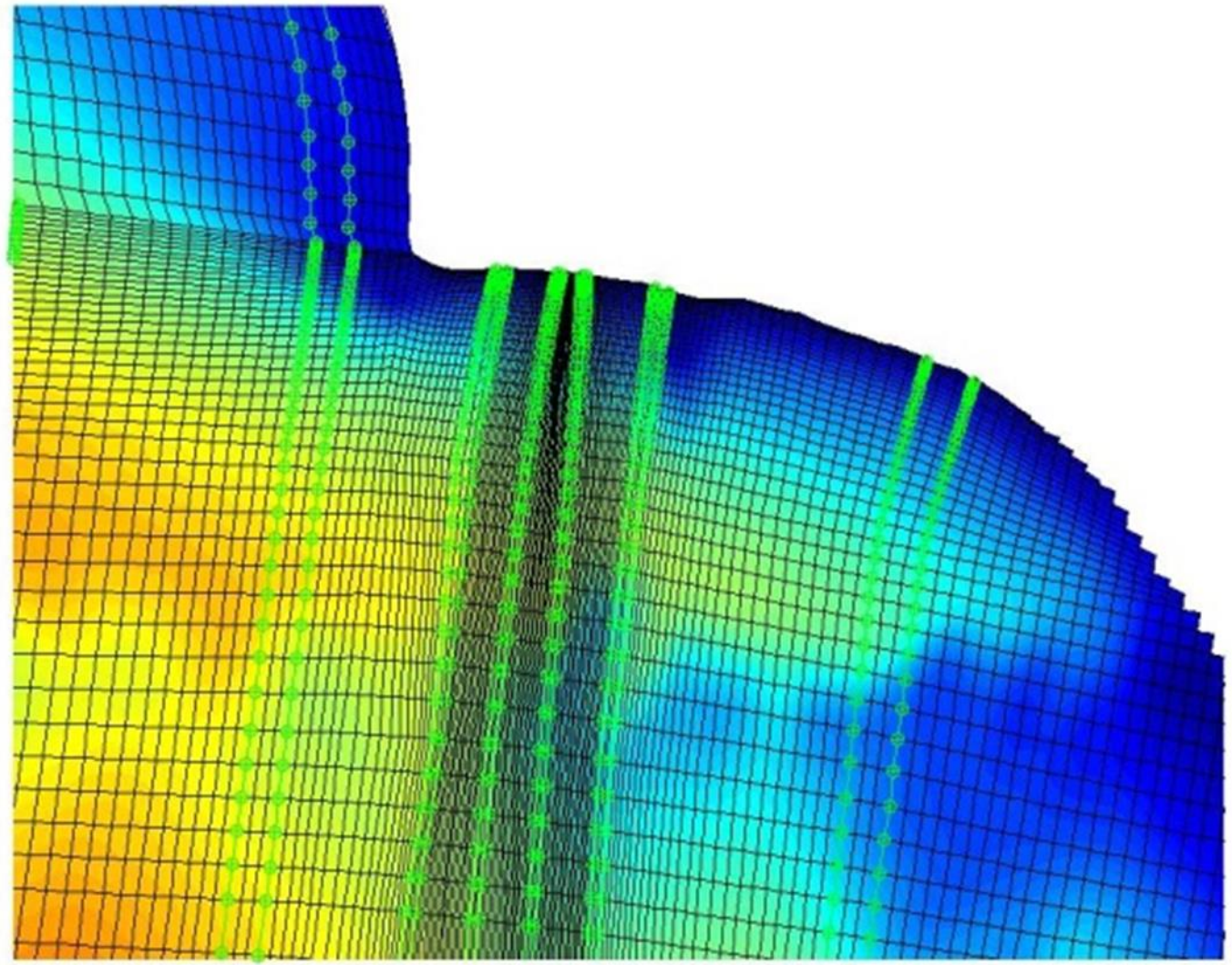


Map showing the ADCIRC model boundary as black lines and the STWAVE boundary as red lines.


Selawik Lake
Kotzebue Sound

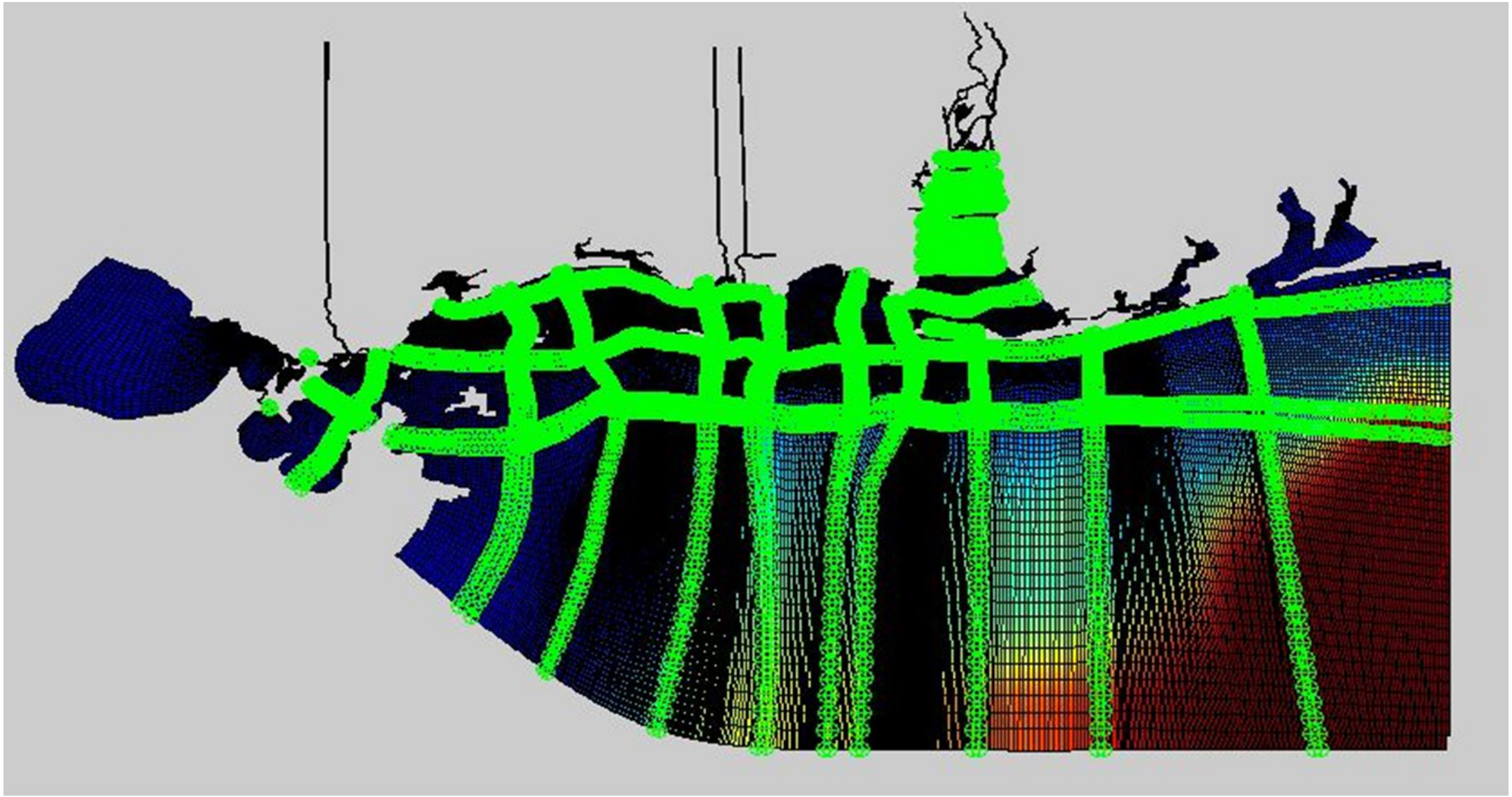


Ten Grid-Block GSMB Model Domain

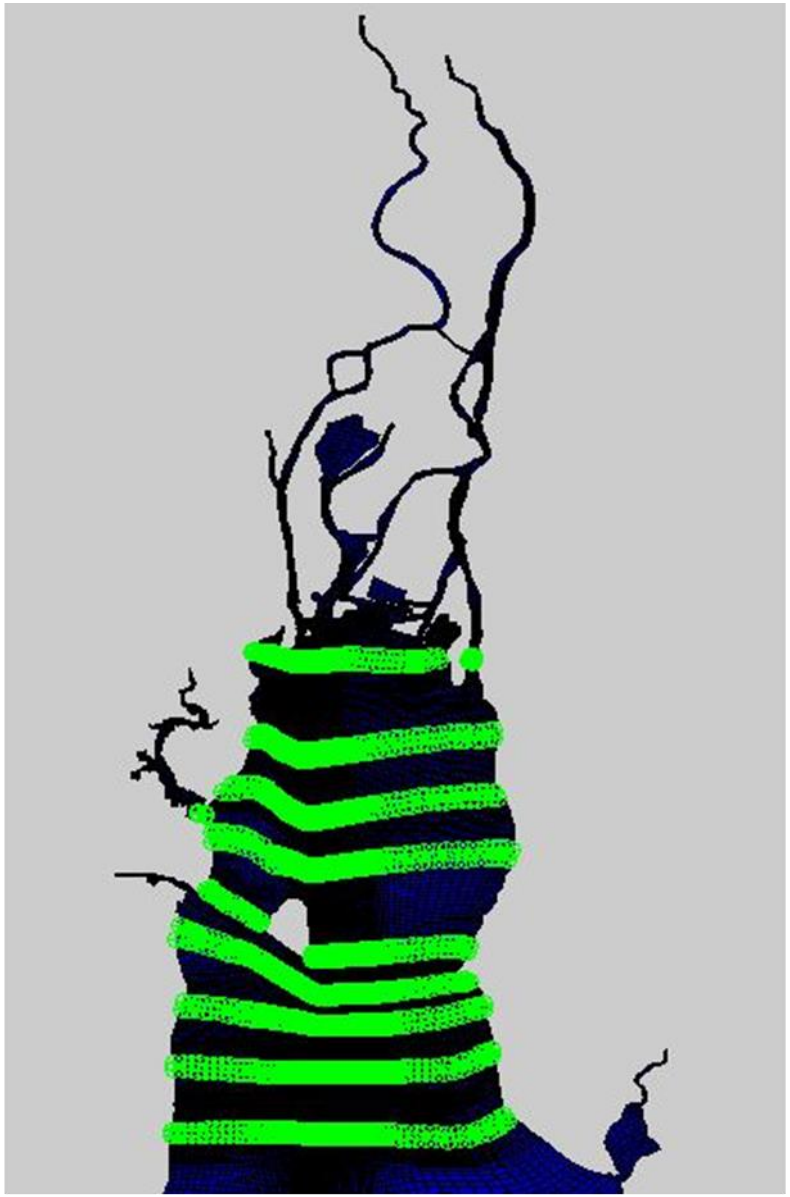


Zoomed in view of Grid-Blocks in Proximity to Cape Blossom

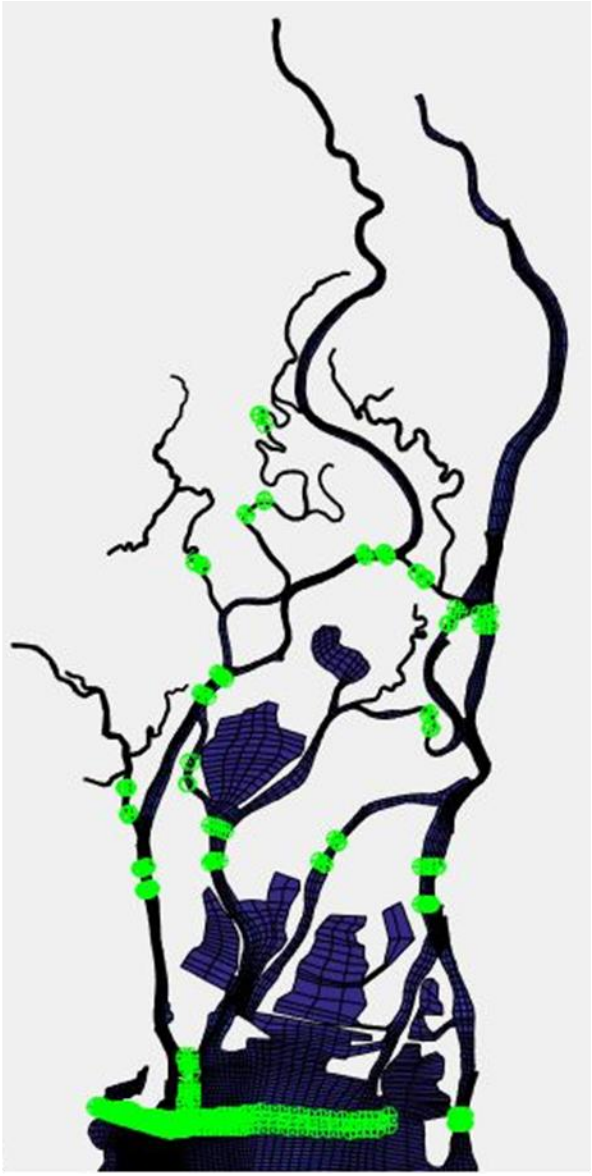
3D Test Case: Three Dimensional Hydrodynamic, Water Quality, and Sediment Transport Modeling of Mobile Bay



Mobile Bay 49 Grid-Block System



Upper Bay and Single Block Delta System



9 Grid-Block Delta System

CORSED: Consolidated Sediment Transport Code

Scope of Work

2. Develop and test the initial CORSED sediment transport library structure shown in Figure 1a. This task will develop a platform-independent library that combines, with different interfaces, two ERDC workhorse sediment transport codes, SEDLIB and SEDZLJ.

   a. Design a new framework (CORSED) for encompassing/connecting the existing standalone sediment libraries SEDLIB and SEDZLJ.
   b. Create CORSED so that SEDLIB and SEDZLJ are both able to exist side-by-side and be built with one unified library called CORSED.
   c. Modify SEDZLJ so that their inputs/outputs and call structures follow the framework standard developed in task 2a.
   d. Develop initial documentation for CORSED.
   e. Per CHL numerical model and technology modernization plan, İmplement Version Control for CORSED.
   f. Perform Verification and Validation (V&V) activities for CORSED.
   g. Develop and document Release and License procedures.
   h. Write Python codes for wrappers to include visualization.
   i. Upgrade the SMS Graphical User Interface (GUI) to include CORSED.

Task 2a: Develop and test the initial CORSED library structure

Common Interface (API) design for wrapping SEDLZJ and SEDLIB

Approach: Use common “STRUCT” objects and arrays purposefully designed to be accessible from C, C++, Fortran and Python codes.

Examples as of Feb 2020:

Grain “STRUCT” as defined by Fortran:

```
type, bind(c) :: type_grain
   integer(kind=c_int) :: type;                ! Should be either "CLA" or "SND"
   integer(kind=c_int) :: index;
   real(kind=c_double) :: diameter;
   real(kind=c_double) :: reference_c;
   real(kind=c_double) :: settling_velocity;
end type type_grain
```

Grain “STRUCT” as defined by C, C++, Python

```
typedef struct
{
   int type;                    // Should be either "CLA" or "SND"
   int index;
   double diameter;             // m
   double reference_c;
   double settling_velocity;    // m/s
} type_grain;
```

Core “STRUCT: as defined by Fortran:

```
type, public, bind(C) :: type_core

  real(kind=c_double), dimension(SEDJ_LAYMAX) :: bdennew                ! (laymax) bulk density

  real(kind=c_double), dimension(SEDJ_LAYMAX,SEDJ_ITBM) :: e0rate   ! (laymax, itbm)  Erosion
        rate for each layer subject to shear stress TAULOC

  real(kind=c_double), dimension(SEDJ_LAYMAX,SEDJ_NM_SED) :: pnew   ! (laymax, nm_sed)
        particle size distribution as a base 100 percentage

  real(kind=c_double), dimension(SEDJ_ITBM) :: tauloc          ! (itbm)   shear stress used to erode a
        portion of the core
  real(kind=c_double), dimension(SEDJ_LAYMAX) :: tautemp  ! (laymax) read the critical shear
        stresses of the core

  real(kind=c_double), dimension(SEDJ_LAYMAX) :: tsed0s    ! [LAYMAX] Initial layer
        thicknesses per core

end type type_core
```

Core “STRUCT” as defined by: C,C++, Python

```
typedef struct
{
   double bdennew[SEDJ_LAYMAX];                    // [LAYMAX]

   double e0rate[SEDJ_ITBM][SEDJ_LAYMAX];      // [NSICM, ITBM]  Erosion rate for each layer
          subject to shear stress TAULOC

   double pnew[SEDJ_NM_SED][SEDJ_LAYMAX];  // [LAYMAX, NM_SED]

   double tauloc[SEDJ_ITBM];                  // [ITBM]  Shear stress used to erode a portion of the core

   double tautemp[SEDJ_LAYMAX];          // [LAYMAX]  Read the critical shear stresses of the core

   double tsed0s[SEDJ_LAYMAX];           // [LAYMAX]  Initial core layer thicknesses

} type_core;
```

CORSED: Consolidated Sediment Transport Code

Scope of Work

3. Develop and test the linkage between SEDZLJ-Lib-v1 and CMS and ADH.
   a. Link SEDZLJ-Lib-v1 to CMS.
   b. Thoroughly test the linkage between SEDZLJ-Lib-v1 and CMS.
   c. Link SEDZLJ-Lib-v1 to ADH.
   d. Thoroughly test the linkage between SEDZLJ-Lib-v1 and ADH.

4. Initiate the development of the second CORSED sediment transport library structure shown in Figure 1b. This structure will permit users to include process descriptions presently in SEDZLJ or SEDLIB as well as include additional process algorithms developed in the future (but not as part of this research).
   a. Modify the initial CORSED sediment transport library developed in Task 2 by re-writing the two libraries to extract common tasks and processes.
   b. Develop a common module to house these common tasks/processes.
   c. Modify CORSED to function as structured in Figure 1b.

All Planned Products/Deliverables

Product Title: Library form of CORSED-SEDZLJ and validation at a test site
Type: **TN**
Status**:** Scheduled Completion Sep 2020

Product Title: Conceptual framework of SEDZLJ, SEDLIB, and CORSED
Type: **TN**
Status: Scheduled Completion Jun 2020

Product Title: CORSED, including all process routines
Type: **TR**
Status: Scheduled Completion Apr 2021

Product Title: CORSED User's Manual
Type: **Living online documentation**
Status: Scheduled Completion Sep 2021

Product Title: CORSED library implemented in ADH, GSMB, and CMS
Type: **Software**
Status: Scheduled Completion Sep 2022

Questions?