An integrated simulation method for coupled dynamic systems

Xu Huang\(^1\) and Oh-Sung Kwon\(^2\)

\(^1\)Ph.D. Candidate, Department of Civil and Mineral Engineering, University of Toronto - Toronto, ON, Canada.
\(^2\)Associate Professor, Department of Civil and Mineral Engineering, University of Toronto - Toronto, ON, Canada.

ABSTRACT

Partitioned analysis methods have been mostly used in multiphysics and large-scale media problems since they allow decomposition of a complex system into smaller problems. Although they have been considered to be superior to monolithic methods in terms of software reuse, difficulties still exist in the implementation process. This study addresses these difficulties and proposes a new method to ease the coupling of the substructures analyzed with different Finite Element (FE) codes. This is enabled by the development of a staggered partitioned approach such that each involved program acts as a black box which is accessible through standard model input and output, i.e. displacements and forces at the boundary of each model. In addition, the substructures can be numerically integrated using any \(\alpha\)-modified integration scheme (explicit or implicit). A general stability proof is provided to determine the maximum time step to ensure a stable solution. An example application will be presented which demonstrates potential of the developed integrated simulation method.

Keywords: Coupled Dynamic System, Substructuring, Integration Scheme, Stability, Finite Element Method

INTRODUCTION

Integrated simulation methods, also termed as partitioned or coupled methods, are most suited for performance assessment of large structural systems as they allow conversion of a large problem into several small problems whose size and complexity can be handled by conventional simulation methods.

Considering a large system whose equation of motion can be expressed as

\[
Ma + Cv + R = F
\]

where \(M\) and \(C\) are system mass and damping matrices, \(R\) and \(F\) are restoring force and external load vectors, \(a\) and \(v\) are acceleration and velocity vectors. There are mainly two types of methods to decompose the equation. For the sake of simplicity, it is assumed that a system is decomposed into two substructures \(i\) and \(j\) which are independently modelled with analysis modules X and Y, respectively.

Method-a: decomposition is only made for the damping and/or restoring forces of the system. Then Eq. 1 can be rewritten as

\[
Ma + C_i v_i + C_j v_j + R_i + R_j = F
\]

where the subscripts \(i\) and \(j\) denote the relevant terms of substructure \(i\) and \(j\), respectively. Figure 1(a) illustrates the way to model the substructures with analysis module X and Y. Including the inertia properties and the external load of the entire system, module X acts as the main solver of the equation of motion (Eq. 2) in addition to the computations of the damping and restoring forces for the substructure \(i\). Module Y, on the other hand, only works as a slave to the module X and its main task is to determine the \(C_j v_j\) and \(R_j\) and provide them to the module X. This method has been mostly used in the current hybrid simulation tests [1,2] where the module Y can be a sophisticated numerical model or a physical specimen. Since the behaviour of the critical portions is often dominated by a few structure components, this method is referred to as a component-level decomposition method.

Method-b: decomposition is not only made for damping and/or restoring forces but also for the inertial properties of the system.

\[
M_i a_i + M_j a_j + C_i v_i + C_j v_j + R_i + R_j = F_i + F_j
\]

The modelling of the two substructures with analysis modules X and Y is shown in Figure 1(b). Different from the master-slave approach used in Method-a, this method equally treats both substructures and dynamic integrations are performed in both
X and Y modules. This method is also known as the domain-decomposition method which is aimed to fully capture the dynamic interaction between different physical fields or domains. To distinct from Method-a, Method-b is referred to as the system-level decomposition method.

![Diagram](image)

(a) Component-level decomposition
(b) System-level decomposition

Figure 1. Component-level vs. system-level decomposition

A generalized method to allow component-level decompositions has been developed by the authors [3]. This study only focuses on the development of system-level decomposition method which involves coupling of dynamic systems.

System-level decompositions are initiated from the domain decomposition methods which are used to solve coupled multiphysics problems, such as fluid-structure interaction and soil-structure interaction. The advent of parallel computing and hybrid simulations extend the method to analyze large-scale or complex systems by dividing the system into small problems which can be solved with different processors or with different approaches [4]. The decomposed subsystems can be solved using either monolithic or partitioned approaches. Based on the definitions in Felippa et al. [5], monolithic methods refer to the ones where the whole problem is treated as a monolithic entity, and all system components advanced simultaneously in time. Although the methods allow each subsystem to formulate its own equation, the response of each subdomain is obtained from the solution of a single governing equation of the entire system considering the contributions from all involved subsystems. Therefore, the interaction effects among the subsystems are directly taken into account. The monolithic methods are also referred to as direct or standard methods [6]. The partitioned methods, on the other hand, treat the problem as isolated entities that are separately stepped in time. The methods involve independent formulations as well as solutions of the governing equations of the subsystems. Therefore, different time integration schemes (or integrators) can be used. The interaction effects between the subsystems are considered by iteratively passing solution variables at the interface from one subsystem to the others until the convergence criterion is achieved [7]. The partitioned methods recently attracted more attention since it overcomes the implementation difficulties in the monolithic methods [8].

![Diagram](image)

Figure 2. Typical staggered solution approach for a two-field problem [5]

The partitioned algorithms can be implemented in a staggered or an iterative manner. In staggered methods, there are two analysis phases for each time step analysis: prediction (P) and substitution (S). As illustrated in Figure 2, each time step analysis starts from one of the substructures, substructure $i$, by introducing a predictor which can be displacement, velocity, acceleration, and force at the interface degrees of freedom (DOFs). Such predictor is then sent to the other substructure, substructure $j$. Once the substructure $j$ is solved, the interface response is substituted into the substructure $i$ such that a new predictor can be generated for the next time step analysis. The method is clearly approximate as most likely the boundary conditions at the interface obtained from the prediction and substitution phases are different. Such difference indicates a violation of either the compatibility or equilibrium condition at the interface, which in turn deteriorates stability and accuracy [9]. Generally, only conditional stability can be achieved when a staggered algorithm is used, and its performance is strongly dependent on the assumption made in the prediction phase. The iterative methods can be viewed as an improved version of the staggered methods to satisfy compatibility and equilibrium at the interface through corrective iterations. Although the iterative methods have been proven to be superior to the conventional staggered methods in accuracy and stability, they are more suited to be implemented in a single program as the methods involve rolling back the analysis for iterations [10–12]. Because iterations are not required, the staggered method is thus considered to be more ideal for reuse of existing code.

The objective of this study is to develop a system-level decomposition method to allow direct coupling of dynamic substructures modelled with different analysis programs. The method is aimed to have the following features:
• Data exchange between the subsystems is only based on numerical model’s standard input/output such that the implementation of the proposed method to existing analysis tools is straightforward. Each involved analysis program is treated as a black box and access to the source code is not necessary.
• Different from many other staggered partitioned methods whose stability characteristics are difficult to be evaluated, a stability proof is available to determine the maximum time step for stable solutions.

COUPLING METHODOLOGY
The system’s equation of motion can be solved numerically using step-by-step integration schemes. Considering the widely used α-modified equation of motion in structural dynamics [13], a linear system can be expressed as

\[ Ma_{n+1} + (1 + \alpha)Cv_{n+1} - \alpha Cv_n + (1 + \alpha)R(d_{n+1}) - \alpha R(d_n) = (1 + \alpha)F_{n+1} - \alpha F_n \]  

(4)

where \( R(d_{n+1}) \) and \( R(d_n) \) are restoring force with respect to \( d_{n+1} \) and \( d_n \), respectively. \( \alpha \) is a coefficient allowing the tuning of the numerical damping to filter out undesired spurious high frequencies. A lower value of \( \alpha \), which is a negative term, leads to a larger numerical damping. The computations of displacement and velocity can be generalized to have a predictor step and a corrector step.

A predictor step:

\[ \dd{d}_{n+1} = d_n + \Delta tv_n + \frac{1-2\beta}{2} \Delta t^2 a_n \]  

(5)
\[ \dd{v}_{n+1} = v_n + (1 - \gamma) \Delta t a_n \]  

(6)

A corrector step:

\[ d_{n+1} = \dd{d}_{n+1} + \beta \Delta t^2 a_{n+1} \]  

(7)
\[ v_{n+1} = \dd{v}_{n+1} + \gamma \Delta t a_{n+1} \]  

(8)

where \( \Delta t \) is time increment, and \( \beta \) and \( \gamma \) are integration coefficients. With specific values of the coefficients, the derived integration scheme can be explicit or implicit. In the predictor step, the nodal displacements and velocities at the time step \( n + 1 \) are explicitly obtained from the state values at the previous time step \( n \). The predicted displacements and velocities are then corrected using the information, i.e. \( a_{n+1} \), at the current time step in the corrector step. Based on Eq. 5 to Eq. 8, Eq. 4 can be transformed into an effective equation as

\[ Aa_{n+1} = B_{n+1} \]  

(9)

with

\[ A = M + (1 + \alpha)\gamma \Delta t C + (1 + \alpha)\beta \Delta t^2 K \]  

(10)
\[ B_{n+1} = (1 + \alpha)F_{n+1} - \alpha F_n - (1 + \alpha)C\dd{v}_{n+1} + \alpha Cv_n - (1 + \alpha)R(\dd{d}_{n+1}) + \alpha R(d_n) \]  

(11)

A generalized form of the effective equation for the substructure \( i \) (sub-\( i \)) interfacing with the substructure \( j \) (sub-\( j \)) can be formulated as

\[ (A_i + A_{ij})a_{i,n+1} = B_{i,n+1} + D_{ij,n+1} \]  

(12)

with

\[ A_i = M_i + (1 + \alpha_i)\gamma_i \Delta t C_i + (1 + \alpha_i)\beta_i \Delta t^2 K_i \]  

(13)
\[ B_{i,n+1} = (1 + \alpha_i)F_{i,n+1} - \alpha_i F_{i,n} - (1 + \alpha_i)C_i\dd{v}_{i,n+1} + \alpha_i Cv_{i,n} - (1 + \alpha_i)R_i(\dd{d}_{i,n+1}) + \alpha_i R_i(d_{i,n}) \]  

(14)

where subscripts \( i \) and \( j \) represent sub-\( i \) and sub-\( j \), respectively. \( A_{ij} \) and \( D_{ij,n+1} \) represent the interaction terms that sub-\( j \) imposes to the sub-\( i \). In this study, the interaction terms are approximated as below.
\[
A_{ij} = \begin{pmatrix} 0 & 0 \\ 0 & M_j^p \end{pmatrix}
\]

\[
D_{ij,n+1} = \begin{pmatrix} 0 \\ (1 + \alpha_i)F_{i,n+1}^B - \alpha_iF_{i,n+1}^B - (1 + \alpha_i)\left(c_j^B\ddot{v}_{i,n+1}^B + \dot{c}_j^B\dot{v}_{i,n+1}^B \right) + \\
\alpha_i\left(c_j^B\dot{v}_{i,n+1}^B + C_j^B\dot{v}_{i,n+1}^B \right) - (1 + \alpha_i)R(\ddot{d}_{i,n+1}) + \\
\alpha_iR(\ddot{d}_{i,n}) \end{pmatrix}
\]

where superscripts \(B\) and \(I\) are used to indicate the terms with respect to the DOFs at the interface boundary between sub-i and sub-j and the rest of the DOFs of sub-j, respectively.

The flow chart of the proposed method is given in Figure 3. The method is apparently staggered as the two substructures are alternatively integrated in time. Step 4 and step 7 in the figure correspond to the prediction and substitution phases in Figure 2, respectively. To distinguish the prediction and substitution phases in the proposed method, the substructure from which \(A_{ij}\) and \(D_{ij,n+1}\) are predicted is referred to as a primary substructure (i.e. sub-j) while the other substructure which receives \(A_{ij}\) and \(D_{ij,n+1}\) is referred to as a secondary substructure (i.e. sub-i). It is worth noting that the proposed method allows different numerical time integration schemes (i.e. different integration coefficients, \(\alpha, \beta, \) and \(\gamma\)) to be used in the substructures.

1. Choose \(\Delta t\)
2. Sub-j predicts \(A_{ij}\) based on Eq. 15 and adds it to the interface boundary of Sub-i
3. Set \(n = 0\)
4. Sub-j predicts \(D_{ij}\) based on Eq. 16 and impose it to the interface boundary of Sub-i
5. Solve Sub-i for \(a_{i,n+1}\) according to Eq. 12
6. Compute \(d_{i,n+1}\) and \(v_{i,n+1}\) of the Sub-i according to Eq. 7 and Eq. 8
7. Sub-i imposes \(d_{i,n+1}^B\) to the interface boundary of the Sub-j.
8. Solve Sub-j according to Eq. 9
9. Set \(n = n+1\) and go to step 4.

**STABILITY ANALYSIS**

The stability of the proposed method is evaluated in this section. Consider a mass-spring system which is decomposed into a primary substructure (sub-j) and a secondary substructure (sub-i) as shown in Figure 4 for example, step 7 in Figure 3 is equivalent to imposing an additional external load at the interface boundary of the primary substructure as shown in Figure 5(b). Since the stability of a numerical integration scheme is independent of the external load, the primary substructure has the same stability characteristics as its counterpart shown in Figure 5(d). In other words, the primary substructure is stable if the stability condition of the integration scheme used in its stability counterpart can be satisfied. Such stability counterpart can be obtained by fixing the interface DOFs of the primary substructure.
For the secondary substructure in Figure 5(a), Eq. 12 can be expanded as

\[
\begin{align*}
\mathbf{M}_i \alpha_{i,n+1} + (1 + \alpha_i) \mathbf{C}_i \mathbf{v}_{i,n+1} - \alpha_i \mathbf{C}_i \mathbf{v}_{j,n} + (1 + \alpha_i) \mathbf{K}_i \mathbf{d}_{i,n+1} - \alpha_i \mathbf{K}_i \mathbf{d}_{i,n} & = (1 + \alpha_i) \mathbf{F}_{i,n+1} - \alpha_i \mathbf{F}_{i,n} \quad (17)
\end{align*}
\]

where

\[
\begin{align*}
\mathbf{M}_i &= \mathbf{M}_i + \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{M}_j - (1 + \alpha_i) \gamma_i \Delta t \mathbf{C}_{j}^{BB} - (1 + \alpha_i) \beta_i \Delta t^2 \mathbf{K}_{j}^{BB} \end{pmatrix} \quad (18) \\
\mathbf{C}_i &= \mathbf{C}_i + \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{C}_{j}^{BB} \end{pmatrix} \quad (19) \\
\mathbf{K}_i &= \mathbf{K}_i + \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{K}_{j}^{BB} \end{pmatrix} \quad (20) \\
\mathbf{F}_{i,n+1} &= \mathbf{F}_{i,n+1} + \begin{pmatrix} 0 \\ \mathbf{F}_{j,n+1} - \mathbf{C}_{j}^{BB} \mathbf{d}_{j,n+1} - \mathbf{K}_{j}^{BB} \mathbf{d}_{j,n+1} \end{pmatrix} \quad (21) \\
\mathbf{F}_{i,n} &= \mathbf{F}_{i,n} + \begin{pmatrix} 0 \\ \mathbf{F}_{j,n} - \mathbf{C}_{j}^{BB} \mathbf{v}_{j,n} - \mathbf{K}_{j}^{BB} \mathbf{d}_{j,n} \end{pmatrix} \quad (22)
\end{align*}
\]

Eq. 17 describes a system with equivalent mass \( \mathbf{M}_i \), equivalent damping \( \mathbf{C}_i \), and equivalent stiffness \( \mathbf{K}_i \). \( \mathbf{F}_{i,n+1} \) and \( \mathbf{F}_{i,n} \) are equivalent external forces which can be neglected for stability analysis. Such system has the same stability characteristics as the original secondary substructure. The system can be obtained by including the mass \( (\mathbf{M}_i^B - (1 + \alpha_i) \gamma_i \Delta t \mathbf{C}_{j}^{BB} - (1 + \alpha_i) \beta_i \Delta t^2 \mathbf{K}_{j}^{BB}) \), stiffness \( \mathbf{K}_{j}^{BB} \) and damping \( \mathbf{C}_{j}^{BB} \) of the primary substructure at the interface boundary to the secondary substructure. For example, the stability counterpart of the secondary substructure of the mass-spring system is shown in Figure 5(c). The equivalent mass \( \bar{m}_i^B \) at the interface node equals to \( m_i + m_j - (1 + \alpha_i) \gamma_i \Delta t c_j - (1 + \alpha_j) \beta_i \Delta t^2 k_j \). If \( \bar{m}_i^B \geq 0 \), the
known stability criterion of the integration scheme used in the stability counterpart can be used to determine the stability of the secondary substructure. For secondary substructures with multiple DOFs at the interface boundary, the equivalent interface mass is shown in Eq. 23 which should be positive definite.

\[
\mathbf{M}_{iB} = \mathbf{M}_{iB}^p + \mathbf{M}_{iB}^s - (1 + \alpha_i)\gamma_i \Delta t \mathbf{C}_{iB}^p - (1 + \alpha_i)\beta_i \Delta t^2 \mathbf{K}_{iB}^p
\]  

(23)

Regardless of the integration schemes used in both primary and secondary substructures, the systems’ stability based on the proposed method can be evaluated by following the steps below:

1. Define stability counterparts of primary and secondary substructures.

2. Find the critical time step \(\Delta t_{cr,s}\) for each stability counterpart based on the known stability criterion of the numerical integration scheme used in the substructure \(s\).

3. The time step \(\Delta t\) selected for the integrated simulation should satisfy

\[
\Delta t \leq \min \{\Delta t_{cr,1}, \Delta t_{cr,2}, \ldots, \Delta t_{cr,h}\}
\]

(24)

where \(h\) is the total number of substructures.

In particular, if unconditionally stable integration schemes are used in both primary and secondary substructures, the stability of the integrated model based on the proposed method is solely dependent on the positive definiteness of the equivalent mass \(\mathbf{M}_{iB}^s\) at interface boundaries of the secondary substructures.

**IMPLEMENTATION**

Based on a staggered scheme, the proposed method facilitates the reuse of existing FE codes for integrated simulations since only boundary quantities, i.e. \(\mathbf{A}_{ij}\) and \(\mathbf{D}_{ij,n+1}\), are required in the coupling process. As shown in Eq. 15, \(\mathbf{A}_{ij}\) is the interface mass of the primary substructure. Since the mass does not change during the analysis in typical structural problems in civil engineering, it can be predefined once at the beginning of the analysis. The force \(\mathbf{D}_{ij,n+1}\) shown in Eq. 16, on the other hand, includes predictors of damping and restoring forces which are difficult to acquire from an analysis program. However, if the primary substructure only includes thin layers of structural elements and they are assumed to be linear elastic, \(\mathbf{D}_{ij,n+1}\) can be computed directly from Eq. 16 based on the predefined damping and stiffness matrices of the layers. For example, an integrated simulation of a gravity dam shown in Figure 6 can be decomposed into three secondary substructure models: the dam structure, the near-field soil domain, and the far-field soil domain. As shown in Figure 7, the decomposition of the system introduces two thin elastic layers of the soil elements modelled in a primary substructure model. Due to the small size of the layers compared to the secondary substructures, the error due to the assumption that the two layers are linear elastic is negligible.

![Figure 6. Geometry of a gravity dam system](image)
APPLICATION EXAMPLE

The gravity dam system shown in Figure 6 was analyzed to demonstrate potential of the proposed method for integrated simulations of large structural systems. The dam had a height of 103 m and a width of 70 m. For the sake of simplicity, the dam was assumed to be linear elastic with an elastic modulus of $3.15 \times 10^{10}$ N/m and a mass density of $2.42 \times 10^3$ kg/m$^3$. The dam was assumed to be built upon a hard rock soil with Young’s modulus, mass density, and Poisson’s ratio of $1.75 \times 10^{10}$ N/m$^2$, $1.8 \times 10^3$ kg/m$^3$, and 0.2, respectively. The primary and shear wave velocities of hard rock were $3.28 \times 10^3$ m/s and $2.01 \times 10^3$ m/s. The system was assumed to be subjected to an incident shear wave as shown in Figure 6. The incident angle was 30° with respect to the horizontal axis. A Ricker wavelet with a maximum acceleration of 5.88 m/s$^2$ and a peak frequency of 1.5 Hz was used to represent the incident wave as shown in Figure 8.

Figure 7 shows the decomposed substructures of the system, each of which was modelled with ABAQUS. Energy absorbing elements were used to mimic the infinite nature of the soil. The incident wave was applied in the far-field soil model as equivalent forces in the horizontal and vertical directions at the bottom boundary of the soil domain. Rayleigh’s damping was used for the entire system with 5% damping specified in the first and second modes. The time step of the analysis was defined as 0.001 sec which was smaller than the critical time step based on the stability analysis method presented in this study.

The results of the integrated model were compared with those of the standalone model where the complete soil-dam system was modelled with ABAQUS. Figure 9 compared the two models in horizontal deflection histories at the two sections (i.e. X-
X and Y-Y) shown in Figure 6. The deflections predicted from the integrated model were in good agreement with those of the standalone model, which confirms the accuracy of the proposed method.

![Figure 6](image_url)

**Figure 6. Predicted horizontal displacement histories of the sections X-X and Y-Y**

**SUMMARY**

This paper presented a simulation method to integrate dynamic systems. The method used a staggered approach to allow data exchange only performed at the interface DOFs between the substructures thereby facilitating software reuse. The method is conditionally stable, and its stability can be evaluated through the stability analysis method developed in this study. Potential of the method for integrated simulations of large structural systems was demonstrated through an application example.

**ACKNOWLEDGMENTS**

The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number CRDPJ 480802-15]

**REFERENCES**


