



OPEN A time-domain Runge-Kutta dual reciprocity boundary element method for scalar wave propagation problem

Fenglin Zhou^{1,2}, Jiehua Deng¹, Junfeng Man¹, Yu Qin³, Jianghong Yu^{3,4}, Yijing Zhang⁵✉ & Wenyuan Li³✉

This paper presents a combined approach utilizing the dual reciprocity boundary element method (DRBEM) and the fourth-order Runge–Kutta method (RKM) to address acoustic radiation problems. Within the dual reciprocity framework, a radial basis function (RBF) is introduced to convert the domain integral into a boundary integral form. Following boundary element discretization, a system of second-order ordinary differential equations (ODEs) is derived. These are subsequently transformed into a set of first-order ODEs (Euler equations) via the introduction of an intermediate variable. The fourth-order Runge–Kutta method is then employed to numerically integrate the resulting system. The validity and accuracy of the proposed method are demonstrated through two numerical examples involving acoustic wave propagation in different structural domains.

Keywords Dual reciprocity boundary element method, Runge-Kutta method, Acoustic radiation, Radial basis function

Acoustic analysis employs a variety of numerical methods, including the Method of Moments (MoM)¹, the boundary element method (BEM)^{2,3}, the plane wave time-domain (PWTD) algorithm⁴, the singular boundary method (SBM)⁵, the finite element method (FEM)^{6,7}, and the finite integration technique (FIT)⁸. Among these, the BEM is particularly well-suited for acoustic problems due to its use of a fundamental solution that inherently satisfies the Sommerfeld radiation condition in unbounded domains. This allows the BEM to efficiently handle infinite domain problems—such as acoustic and electromagnetic fields—with discretization required only on the boundary, often yielding higher accuracy with lower computational cost compared to domain-type methods⁹.

BEM implementations can be categorized into frequency-domain (FDBEM)¹⁰ and time-domain (TDBEM)^{11,12} approaches. In FDBEM, the governing equation is transformed into the Helmholtz equation, which is then converted into a boundary integral equation using a frequency-dependent fundamental solution. While efficient for single-frequency problems, FDBEM requires repeated matrix inversions for broadband analyses, which becomes computationally expensive for large-scale problems.

In TDBEM, there are two different kinds of implementations. The one is to utilize the time-dependent fundamental solution¹³. Although the domain integral could be avoided in the boundary integral equation, a time consuming convolution quadrature is usually involved. Moreover, the calculation of boundary integral becomes very difficult due to the causality of the fundamental solution. The other one is to utilize the time-independent fundamental solution to the Laplace equation. This introduction of time-independent fundamental solution leads to a domain integral term which is related to the sound pressure time variation in the conventional boundary integral equation. To avoid direct domain integration, several techniques have been developed, such as the dual reciprocity method (DRM)¹⁴, the radial integration method (RIM)¹⁴, and the multiple reciprocity method (MRM)¹⁵. The DRM and RIM, which utilize radial basis function (RBF) interpolation to approximate the domain term, are widely adopted as they preserve the boundary-only character of the BEM. The DRBEM has

¹School of Intelligent Manufacturing, Hunan First Normal University, Changsha 410205, China. ²Key Laboratory of Industrial Equipment Intelligent Perception and Maintenance Technology in College of Hunan Province, Hunan First Normal University, Changsha 410205, China. ³School of Mechanical Engineering, Hunan University of Technology, Zhuzhou 412001, China. ⁴Key Laboratory of High-Performance Rolling Bearings in Hunan Province, Zhuzhou 412001, China. ⁵Collaborative Innovation Center, Hunan Automotive Engineering Vocational University, Zhuzhou 412001, China. ✉email: 275391827@qq.com; 2321849076@qq.com

been successfully applied to nonlinear wave problems¹⁶, transient convection–diffusion–reaction problems¹⁷, transient heat conduction^{18,19}, viscoelastic fracture problems²⁰, among others.

In acoustic applications involving mechanical vibrations or shocks, the low-frequency range (0–200 Hz) dominates the spectrum, making time-domain methods particularly relevant. After DRM discretization of the time-domain wave equation, a system of semi-discrete second-order ODEs is obtained. Traditional time-stepping schemes often suffer from numerical instability²¹. The Runge–Kutta method (RKM), known for its high convergence rate and stability^{22,23}, is well-suited for such systems.

This paper employs a fourth-order RKM to solve the ODE system derived from the DRM discretization. Since the RKM is designed for first-order systems, the second-order ODEs are converted into first-order form by introducing an auxiliary variable, effectively doubling the system size.

The remainder of this paper is organized as follows: “The dual reciprocity boundary element method for scalar wave propagation” outlines the DRBEM formulation for scalar wave propagation. Section “Dual reciprocity boundary element method for scalar wave propagation problem” details the application of the RKM to the semi-discretized ODE system. Section “Numerical example” presents two numerical examples validating the proposed method. Conclusions and future research directions are discussed in “Conclusion”.

The dual reciprocity boundary element method for scalar wave propagation The governing equation of the acoustic scattering problem

In an ideal homogeneous domain Ω which is enclosed by the boundary Γ , the acoustic propagation in the absence of internal sound sources can be described by the following equation

$$\Delta u(\mathbf{x}, t) = \frac{1}{c^2} \ddot{u}(\mathbf{x}, t) \quad \mathbf{x} \in \Omega \quad (1)$$

where c denotes the speed of sound propagation, the superscript $\ddot{}$ denotes the second-order derivative with respect to time variable t , $u(\mathbf{x}, t)$ denotes the sound pressure at the point \mathbf{x} in the medium at the time t , and Δ is the Laplace operator, which is expressed in two-dimensional as well as three-dimensional models as follows

$$\begin{cases} \Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} & 2D \\ \Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \frac{\partial^2}{\partial x_3^2} & 3D \end{cases} \quad (2)$$

There are three types of boundary conditions

$$\begin{cases} u(\mathbf{x}, t) = \bar{u}(\mathbf{x}, t) & \mathbf{x} \in \Gamma_1 \\ \frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}, t) = \bar{u}_n(\mathbf{x}, t) & \mathbf{x} \in \Gamma_2 \\ \alpha u(\mathbf{x}, t) + \beta \frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}, t) = 0 & \mathbf{x} \in \Gamma_3 \end{cases} \quad (3)$$

In which, \mathbf{n} stands for the outward normal to the corresponding surface. To guarantee the existence of the solution, the initial condition as follows should be given.

$$\begin{aligned} u(\mathbf{x}, t_0) &= u_0(\mathbf{x}) \quad \mathbf{x} \in \Omega \\ \frac{\partial u}{\partial t}(\mathbf{x}, t_0) &= \dot{u}_0(\mathbf{x}) \quad \mathbf{x} \in \Omega \end{aligned} \quad (4)$$

where $\Gamma_1, \Gamma_2, \Gamma_3$ are the Dirichlet boundary, the Neumann boundary and the Robin boundary, respectively. They satisfy $\Gamma_1 \cup \Gamma_2 \cup \Gamma_3 = \Gamma$.

Dual reciprocity boundary element method for scalar wave propagation problem

The governing equation Eq. (1) can be converted into an equivalent integral equation by employing a fundamental solution to the Laplacian.

$$\begin{aligned} C(\mathbf{y})u(\mathbf{y}, t) &= \int_{\Gamma} u(\mathbf{x}, t) \left(\frac{\partial u^*(\mathbf{y}, \mathbf{x})}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) - \int_{\Gamma} u^*(\mathbf{y}, \mathbf{x}) \left(\frac{\partial u(\mathbf{x}, t)}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) \\ &\quad - c^2 \int_{\Omega} u^*(\mathbf{y}, \mathbf{x}) \frac{\partial^2 u(\mathbf{x}, t)}{\partial t^2} d\Omega(\mathbf{x}) \end{aligned} \quad (5)$$

where $C(\mathbf{y})$ is the coefficient, $C(\mathbf{y}) = 0$ if \mathbf{y} locates outside the domain, $C(\mathbf{y}) = 1$ if \mathbf{y} locates inside the domain, $C(\mathbf{y}) = 1/2$ if \mathbf{y} locates on a smooth boundary. $u^*(\mathbf{y}, \mathbf{x})$ is the fundamental solution to the Laplacian operator

$$u^*(\mathbf{y}, \mathbf{x}) = \frac{1}{4\pi r(\mathbf{y}, \mathbf{x})} \quad (6)$$

In this conversion, the term is treated as an equivalent heat source which is variable along with time. Thus, the equivalent heat source appears in the right hand side of Eq. (5) in the form of domain integral. In the DRM, the term of equivalent heat source can be approximated through a radial basis function interpolation method with variable coefficients.

$$\frac{\partial^2}{\partial t^2} u(\mathbf{x}, t) = \sum_i \alpha_i(t) \varphi_i(\|\mathbf{x} - \mathbf{x}_i\|) \tag{7}$$

where \mathbf{x}_i is the interpolation center which located inside the considered domain, $\alpha_i(t)$ is the RBF coefficient. There are many types of RBFs involving the Multiquadrics, the Gaussian RBF and the compact supported RBF. The Multiquadrics which is listed in Eq. (8) usually gives a more accurate result than other types of RBF.

$$\varphi_i(r) = \sqrt{r^2 + a^2} \tag{8}$$

where a is the shape parameter whose evaluation plays a very important role in the interpolation.

After the integration by parts for twice from the term $c^2 \int_{\Omega} u^*(\mathbf{y}, \mathbf{x}) \frac{\partial^2 u(\mathbf{x}, t)}{\partial t^2} d\Omega(\mathbf{x})$ in Eq. (5), the following formula can be obtained.

$$\begin{aligned} & C(\mathbf{y})u(\mathbf{y}, t) - \int_{\Gamma} u(\mathbf{x}, t) \left(\frac{\partial u^*(\mathbf{y}, \mathbf{x})}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) \\ & + \int_{\Gamma} u^*(\mathbf{y}, \mathbf{x}) \left(\frac{\partial u(\mathbf{x}, t)}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) \\ & = c^2 C(\mathbf{y}) \sum_i \alpha_i(t) \phi_i(\mathbf{y}) \\ & - c^2 \sum_i \int_{\Gamma} \phi_i(\mathbf{x}) \left(\frac{\partial u^*(\mathbf{y}, \mathbf{x})}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) \alpha_i(t) \\ & + c^2 \sum_i \int_{\Gamma} u^*(\mathbf{y}, \mathbf{x}) \left(\frac{\partial \phi_i(\mathbf{x})}{\partial n(\mathbf{x})} \right) d\Gamma(\mathbf{x}) \alpha_i(t) \end{aligned} \tag{9}$$

where $\phi_i(\mathbf{y})$ is the particular solution and satisfies

$$\nabla^2 \phi_i(\|\mathbf{y} - \mathbf{y}_i\|) = \varphi_i(\|\mathbf{y} - \mathbf{y}_i\|) \tag{10}$$

Within the RBF interpolation, the coefficients can be evaluated through

$$\alpha(t) = \Phi^{-1} \frac{\partial^2}{\partial t^2} \mathbf{u}(t) \tag{11}$$

where $\Phi_{ji} = \varphi(\|\mathbf{x}_j - \mathbf{x}_i\|)$.

In this application, the scattered nodes are generated inside the considered domain. This nodes are employed to perform the RBF interpolation. The Eq. (9) could be extended to the following formula by considering the particular solution.

$$\begin{aligned} & C(\mathbf{y})u(\mathbf{y}, t) \\ & = \sum_j \sum_m \int_{\Gamma_j} \frac{\partial}{\partial n} u^*(\mathbf{y}, \mathbf{x}) u(\mathbf{x}_m, t) N_m(\mathbf{x}) d\Gamma(\mathbf{x}) \\ & - \sum_j \sum_m \int_{\Gamma_j} u^*(\mathbf{y}, \mathbf{x}) \frac{\partial}{\partial n} u(\mathbf{x}_m, t) N_m(\mathbf{x}) d\Gamma(\mathbf{x}) \\ & - c^2 \sum_i \sum_j \sum_m \int_{\Gamma_j} \frac{\partial}{\partial n} u^*(\mathbf{y}, \mathbf{x}) \phi_i(\mathbf{x}_m) N_m(\mathbf{x}) d\Gamma(\mathbf{x}) \alpha_i(t) \\ & + c^2 \sum_i \sum_j \sum_m \int_{\Gamma_j} u^*(\mathbf{y}, \mathbf{x}) \left(\frac{\partial \phi_i(\mathbf{x}_m)}{\partial n(\mathbf{x})} \right) N_m(\mathbf{x}) d\Gamma(\mathbf{x}) \alpha_i(t) \\ & + c^2 C(\mathbf{y}) \sum_i \alpha_i(t) \phi_i(\mathbf{y}) \end{aligned} \tag{12}$$

After collocating the field points at all RBF centers which are distributed inside domain, the following systems could be obtained.

$$\mathbf{H}^{db} \mathbf{u}^b(t) - \mathbf{u}^d(t) - \mathbf{G}^{db} \frac{\partial \mathbf{u}^b(t)}{\partial n} = \left(\mathbf{H}^{bd} \hat{\mathbf{U}} - \hat{\mathbf{U}}^{dd} - \mathbf{G}^{bd} \hat{\mathbf{Q}} \right) \Phi^{-1} \hat{\mathbf{u}}^d(t) \tag{13}$$

in which, d and b denote the number of internal nodes and the number of boundary nodes, respectively. The superscript indicates the size of the corresponding matrix and vectors.

If we collocate the field points at all boundary nodes which locate on smooth boundary, we have

$$\left(\mathbf{H}^{bb} - 0.5 \mathbf{I} \right) \mathbf{u}^b(t) - \mathbf{G}^{bb} \frac{\partial \mathbf{u}^b(t)}{\partial n} = \left(\left(\mathbf{H}^{bb} - 0.5 \mathbf{I} \right) \hat{\mathbf{U}}^{bd} - \mathbf{G}^{bd} \hat{\mathbf{Q}} \right) \Phi^{-1} \hat{\mathbf{u}}^d(t) \tag{14}$$

where \mathbf{I}^{bb} denotes the unit matrix with b rows and b columns (Eq. (14) is only applicable when collocation points are located on smooth boundaries). We can rewrite the Eq. (14) in a larger matrix form as

$$\begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \end{bmatrix} \begin{bmatrix} \mathbf{u}^b(t) \\ \frac{\partial \mathbf{u}^b(t)}{\partial n} \end{bmatrix} = \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}^{bd} \\ \hat{\mathbf{Q}}^{bd} \end{bmatrix} \Phi^{-1} \hat{\mathbf{u}}^d(t) \tag{15}$$

Considering three types of boundary condition, the boundary condition can be described in an uniform expression

$$\beta_1 u(\mathbf{x}, t) + \beta_2 \frac{\partial u(\mathbf{x}, t)}{\partial n(\mathbf{x})} = \beta_3 \mathbf{x} \in \Gamma \tag{16}$$

where

$$\begin{cases} \beta_1 = 1, \beta_2 = 0 & \text{Dirichlet boundary} \\ \beta_1 = 0, \beta_2 = 1 & \text{Neumann boundary} \\ \beta_1 \neq 0, \beta_2 \neq 0 & \text{Robin boundary} \end{cases} \tag{17}$$

The sound pressure and its normal derivative at the boundary nodes should satisfy

$$\begin{bmatrix} \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix} \begin{bmatrix} \mathbf{u}^b(t) \\ \frac{\partial \mathbf{u}^b(t)}{\partial n} \end{bmatrix} = \begin{bmatrix} \beta_3(t) \\ \beta_3(t) \end{bmatrix} \tag{18}$$

where $\mathbf{B}_1^{bb} = \beta_1 \mathbf{I}^{bb}$ and $\mathbf{B}_2^{bb} = \beta_2 \mathbf{I}^{bb}$ are diagonal matrix.

Combining Eq. (15) and Eq. (18) yields

$$\begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix} \begin{bmatrix} \mathbf{u}^b(t) \\ \frac{\partial \mathbf{u}^b(t)}{\partial n} \end{bmatrix} = \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}^{bd} \\ \hat{\mathbf{Q}}^{bd} \end{bmatrix} \Phi^{-1} \hat{\mathbf{u}}^d(t) + \begin{bmatrix} 0 \\ \beta_3(t) \end{bmatrix} \tag{19}$$

Thus, the boundary quantities can be computed through

$$\begin{bmatrix} \mathbf{u}^b(t) \\ \frac{\partial \mathbf{u}^b(t)}{\partial n} \end{bmatrix} = \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}^{bd} \\ \hat{\mathbf{Q}}^{bd} \end{bmatrix} \Phi^{-1} \hat{\mathbf{u}}^d(t) + \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \beta_3(t) \end{bmatrix} \tag{20}$$

If we denote

$$\mathbf{Z}^{2b \times d} = \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}^{bd} \\ \hat{\mathbf{Q}}^{bd} \end{bmatrix} \tag{21}$$

$$\mathbf{e}^{2b} = \begin{bmatrix} \mathbf{H}^{bb} - 0.5 \mathbf{I}^{bb} & -\mathbf{G}^{bb} \\ \mathbf{B}_1^{bb} & \mathbf{B}_2^{bb} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \beta_3(t) \end{bmatrix} \tag{22}$$

Considering Eqs. (20–22), the Eq. (13) can be simplified into

$$\begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \left(\mathbf{Z} \mathbf{F}^{-1} \overset{d}{\mathbf{u}} + \mathbf{e} \right) - \overset{d}{\mathbf{u}} = \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \begin{bmatrix} bd \\ \tilde{\mathbf{U}} \\ bd \\ \tilde{\mathbf{Q}} \end{bmatrix} - \overset{dd}{\tilde{\mathbf{U}}} \right\} \mathbf{F}^{-1} \overset{d}{\mathbf{u}} \quad (23)$$

The term of $\overset{d}{\mathbf{u}}$ can be explicitly expressed by

$$\begin{aligned} \overset{d}{\mathbf{u}} &= \overset{dd}{\Phi} \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \mathbf{Z} - \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \begin{bmatrix} bd \\ \tilde{\mathbf{U}} \\ bd \\ \tilde{\mathbf{Q}} \end{bmatrix} - \overset{dd}{\tilde{\mathbf{U}}} \right\}^{-1} \overset{d}{\mathbf{u}} \right. \\ &\quad \left. - \overset{dd}{\Phi} \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \mathbf{Z} - \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \begin{bmatrix} bd \\ \tilde{\mathbf{U}} \\ bd \\ \tilde{\mathbf{Q}} \end{bmatrix} - \overset{dd}{\tilde{\mathbf{U}}} \right\}^{-1} \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \mathbf{e} \right. \end{aligned} \quad (24)$$

we further denote

$$\overset{dd}{\mathbf{A}} = \overset{dd}{\Phi} \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \mathbf{Z} - \left\{ \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \begin{bmatrix} bd \\ \tilde{\mathbf{U}} \\ bd \\ \tilde{\mathbf{Q}} \end{bmatrix} - \overset{dd}{\tilde{\mathbf{U}}} \right\}^{-1} \right\}^{-1} \quad (25)$$

$$\overset{d \times (2b)}{\mathbf{B}} = - \overset{dd}{\mathbf{A}} \begin{bmatrix} db & -db \\ \mathbf{H} & -\mathbf{G} \end{bmatrix} \quad (26)$$

The following system of second-order ODEs can be derived

$$\overset{d}{\ddot{\mathbf{u}}} = \overset{dd}{\mathbf{A}} \overset{d}{\dot{\mathbf{u}}} + \overset{d \times (2b)}{\mathbf{B}} \overset{d}{\mathbf{e}} \quad (27)$$

Before the implementation of the Runge–Kutta method, an intermedia variable should be introduced to lower the order of the differential Eq.

$$\overset{d}{\dot{\mathbf{u}}} = \overset{d}{\mathbf{v}} \quad (28)$$

Equation (27) is transformed into a first-order ordinary differential Eq.

$$\begin{bmatrix} \overset{d}{\dot{\mathbf{v}}} \\ \overset{d}{\dot{\mathbf{u}}} \end{bmatrix} = \begin{bmatrix} 0 & \overset{dd}{\mathbf{A}} \\ \overset{dd}{\mathbf{I}} & 0 \end{bmatrix} \begin{bmatrix} \overset{d}{\mathbf{v}} \\ \overset{d}{\mathbf{u}} \end{bmatrix} + \begin{bmatrix} \overset{d \times (2b)}{\mathbf{B}} \\ 0 \end{bmatrix} \overset{2b}{\mathbf{e}} \quad (29)$$

Equation (29) can be given by

$$\overset{2d}{\dot{\mathbf{x}}} = \overset{2d \times 2d}{\mathbf{A}} \overset{2d}{\mathbf{x}} + \overset{2d \times (2b)}{\mathbf{B}} \overset{2b}{\mathbf{e}} \quad (30)$$

where $\overset{2d}{\mathbf{x}} = \begin{bmatrix} \overset{d}{\mathbf{v}} \\ \overset{d}{\mathbf{u}} \end{bmatrix}$, $\overset{2d \times 2d}{\mathbf{A}} = \begin{bmatrix} 0 & \overset{dd}{\mathbf{A}} \\ \overset{dd}{\mathbf{I}} & 0 \end{bmatrix}$, $\overset{2d \times (2b)}{\mathbf{B}} = \begin{bmatrix} \overset{d \times (2b)}{\mathbf{B}} \\ 0 \end{bmatrix}$.

Fourth-order Runge-Kutta method

The Runge-Kutta method is widely applied in ODEs solution for its high convergence rate. Although, the high order Runge-Kutta method could be applied directly to solve the second order ODEs²⁴, this direct solution usually involves more parameters than that involved in the reduced-order application. Moreover, the direct solution is considered to be equivalent to the reduced order method essentially. Thus, the Runge-Kutta method is applied to solvereduced order model Eq. (30) rather than Eqs. (27) in this application.

Take the first-ordered linear ordinary differential equation as an example to illustrate the general form of the Runge-Kutta formula

$$\begin{cases} \frac{dx}{dt} = f(t, x) \\ x(t_0) = x_0 \end{cases} \quad (31)$$

where $f(t, x)$ is a known function, t is the time variable and t_0 stands for the initial time.

The Runge-Kutta method gives the recurrence scheme as follow

$$\begin{cases} x_{n+1} = x_n + h \sum_{i=1}^s \omega_i k_i \\ k_i = f(t_n + a_i h, x_n + h \sum_{j=1}^{s-1} \beta_{ij} k_j) \end{cases} \tag{32}$$

where $\omega_i, a_i, \beta_{ij}$ are constants, s is the order of the method, h stands for the step length. A typical fourth-order Runge-Kutta formula is

$$\begin{cases} x_{n+1} = x_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = f(t_n, x_n) \\ k_2 = f(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_1) \\ k_3 = f(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_2) \\ k_4 = f(t_n + h, x_n + hk_3) \end{cases} \tag{33}$$

Substituting Eq. (30) into Eq. (33) yields to the following system of Eq.

$$\begin{cases} x(t_{k+1}) = x(t_k) + \frac{h}{6} \left(K_1 + 2K_2 + 2K_3 + K_4 \right) \\ K_1 = \mathbf{A} x(t_k) + \mathbf{B} e^{2b} (t_k) \\ K_2 = \mathbf{A} \left[x(t_k) + \frac{h}{2} K_1 \right] + \mathbf{B} e^{2b} (t_k + \frac{h}{2}) \\ K_3 = \mathbf{A} \left[x(t_k) + \frac{h}{2} K_2 \right] + \mathbf{B} e^{2b} (t_k + \frac{h}{2}) \\ K_4 = \mathbf{A} \left[x(t_k) + h K_3 \right] + \mathbf{B} e^{2b} (t_k + h) \end{cases} \tag{34}$$

Numerical example

In this section, the Runge-Kutta dual reciprocity boundary element method for scalar wave propagation problems is tested through two numerical examples. A cylinder shaped domain and a cube domain are considered in these two examples, respectively. Within in these two examples, the results obtained by the proposed method are compared with the analytical solution and the result from the precise intergal dual reciprocity method¹⁸, respectively. In both examples, all the physical quantities are dimensionless. The shape parameter of the RBF is evaluated by 0.2 to guarantee the numerical stability of the methods, although larger value of shape parameter usually leads to a more accurate result as suggested in Huang’s work²⁵. The relative errors are computed from the following formulation.

$$error = \frac{1}{|u|_{\max}} \sqrt{\frac{1}{N} \sum_{i=1}^N (u_i^{(refer)} - u_i^{(n)})^2} \tag{35}$$

In which $u_i^{(refer)}$ is the referred value of the corresponding variable at the i -th sample point, $u_i^{(n)}$ stands for the computed value by the presented method, $|u|_{\max}$ denotes the maximum value of the variable at all sampling points.

Scalar wave propagation in a cylindrical tunnel

A cylindrical tunnel with a diameter of 2 and a height of 5 as illustrated in Fig. 1 is considered in this example. The coordinate system is originated at the center of the bottom surface. The axis coincides with the z -direction.

In this implementation, 60 discontinuous serendipity boundary element with in total 231 boundary nodes are employed to discretized the boundary. 488 RBF interpolation points are distributed inside the cylinder to perform the RBF interpolation. The boundary elements and the scattered RBF interpolation points are illustrated in Figs. 2 and 3, respectively.

The distribution and variation of the sound pressure on the boundary is given through the following formula.

$$u(x, y, z, t) = 2x^3 + y^3 + 2.5z^3 + 6xt^2 + 3yt^2 + 7.5zt^2 (x, y, z) \in \Gamma_1, t > 0$$

$$\frac{\partial u}{\partial \mathbf{n}}(x, y, z, t) = 7.5z^2 + 7.5t^2 (x, y, z) \in \Gamma_2, t > 0$$

The initial distribution of sound pressure in the domain is given by.

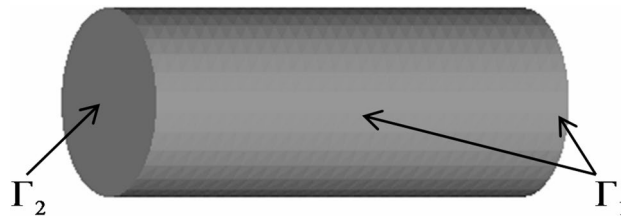


Fig. 1. The considered cylinder domain.

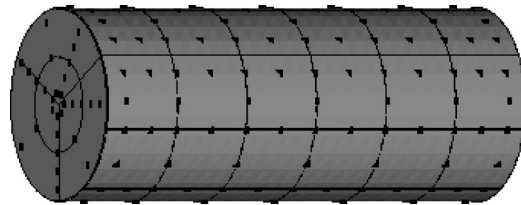


Fig. 2. Cylinder boundary element and node distribution.

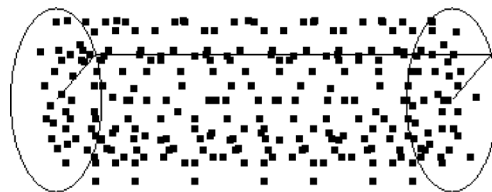


Fig. 3. RBF interpolation point distribution in the cylinder.

Number	Locations (x, y, z)
P1	(0.2392, -0.5774, 4.792)
P2	(-0.613, -0.1219, 4.479)
P3	(0.1294, -0.483, 2.625)
P4	(-0.3437, -0.5954, 1.75)
P5	(-0.613, -0.1219, 0.8333)

Table 1. The sampling points in the first example.

$$u(x, y, z) = 2x^3 + y^3 + 2.5z^3(x, y, z) \in \Omega$$

The exact distribution of the pressure in the domain could be described by the following formula.

$$u(x, y, z, t) = 2x^3 + y^3 + 2.5z^3 + 6xt^2 + 3yt^2 + 7.5zt^2(x, y, z) \in \Omega$$

Five sample points as listed in Table 1 are introduced in this example to illustrate the accuracy of the presented method.

The computational time starts from 0s and ends at 20s. The time step length of 0.008s is applied in this example. The pressure variation history at 5 sample points are illustrated in Fig. 4. The numerical results coincides well with the analytical solution.

To make the comparison more clearly, the pressure at the 2nd sample point and at the 4th sample point in sample moment is listed in Table 2. From this table, it is easy to find that the pressure result computed through the presented method matches nicely with the referred solution.

To further study the convergence of the presented method, 6 different time steps including 0.008, 0.00625, 0.005, 0.004, 0.0025 and 0.002 have been tested in this example. As illustrated in Fig. 5; Table 3, the average relative error at the end of the computed time interval converges stably.

From Table 3; Fig. 5, it can be concluded that the smaller the step size, the smaller sized time step could achieve more accurate result. In this application, the error comes mainly from the spatial discretization.

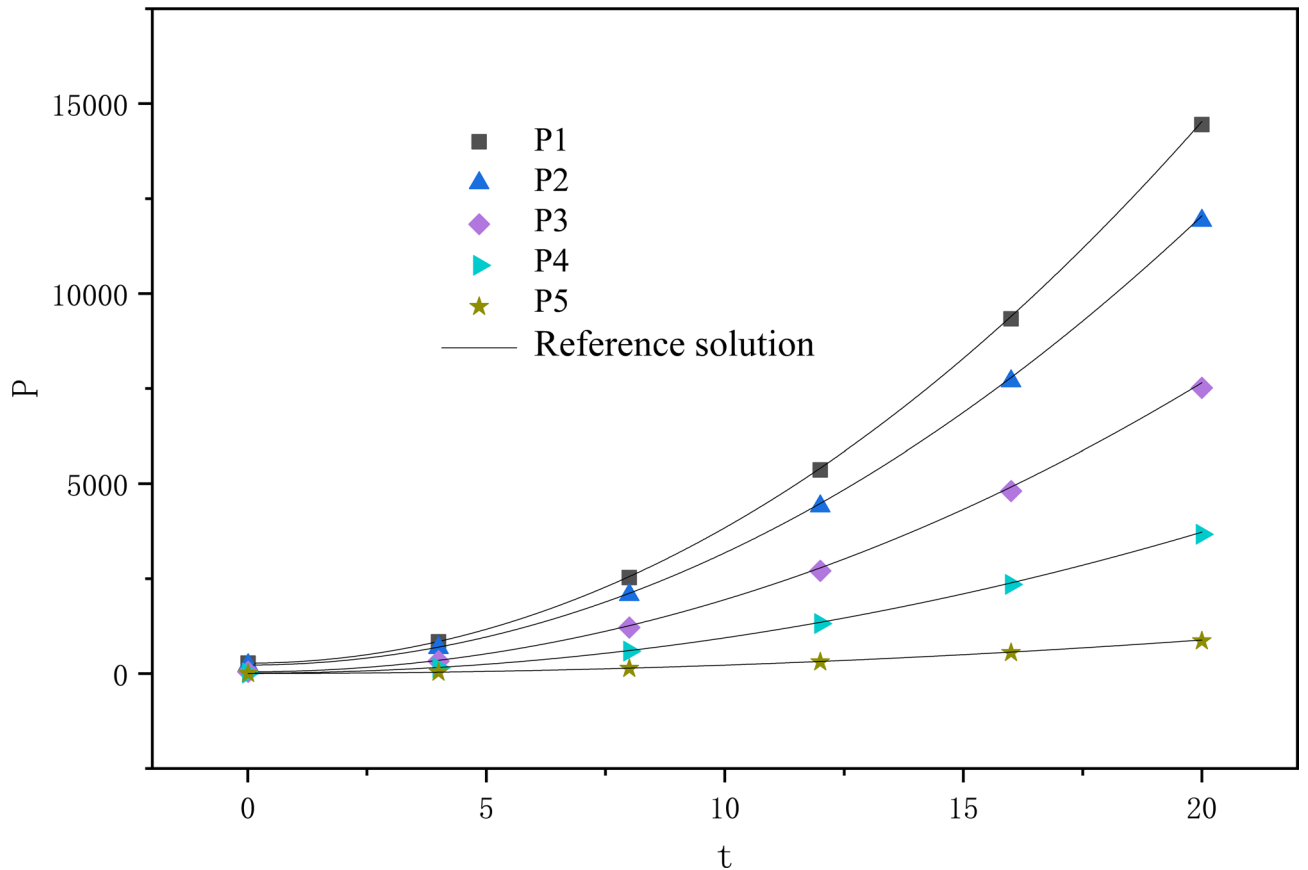


Fig. 4. Calculation results and reference values of sound pressure amplitude at observation points.

Time	P2	P2 reference value	P4	P4 reference value
4	695.615	697.001	158.099	161.527
8	2099.3	2115.4	596.509	606.791
12	4406.65	4479.4	1316.8	1348.9
16	7689.68	7789	2343.96	2387.84
20	11918.4	12044.2	3668.01	3723.64

Table 2. Comparison on pressure results between the RKDRM and the referring method.

In the next example, the Runge-Kutta dual reciprocity boundary element method will be further investigated and compared with the precise integration dual reciprocity boundary element method.

Wave propagation in a cubic region

In this example, a cubic domain occupying in $[0, 1] \times [0, 1] \times [0, 1]$ is considered. This example is presented to illustrate the efficiency of the RKDRM. The result obtained by the RKDRM is compared with that from the precise integral dual reciprocity method (PIDRM).

In this analysis, 54 serendipity elements involving 240 boundary nodes and 64 RBF interpolation nodes as shown in Figs. 6 and 7 are applied.

The boundary pressure is given as:

$$u(x, y, z, t) = 2x^3 + y^3 + 2.5z^3 + 6xt^2 + 3yt^2 + 7.5zt^2 \quad (x, y, z) \in \Gamma, t > 0$$

The initial distribution of the pressure in the domain can be described as:

$$u(x, y, z) = 2x^3 + y^3 + 2.5z^3 \quad (x, y, z) \in \Omega$$

The exact solution for the sound pressure in the domain is.

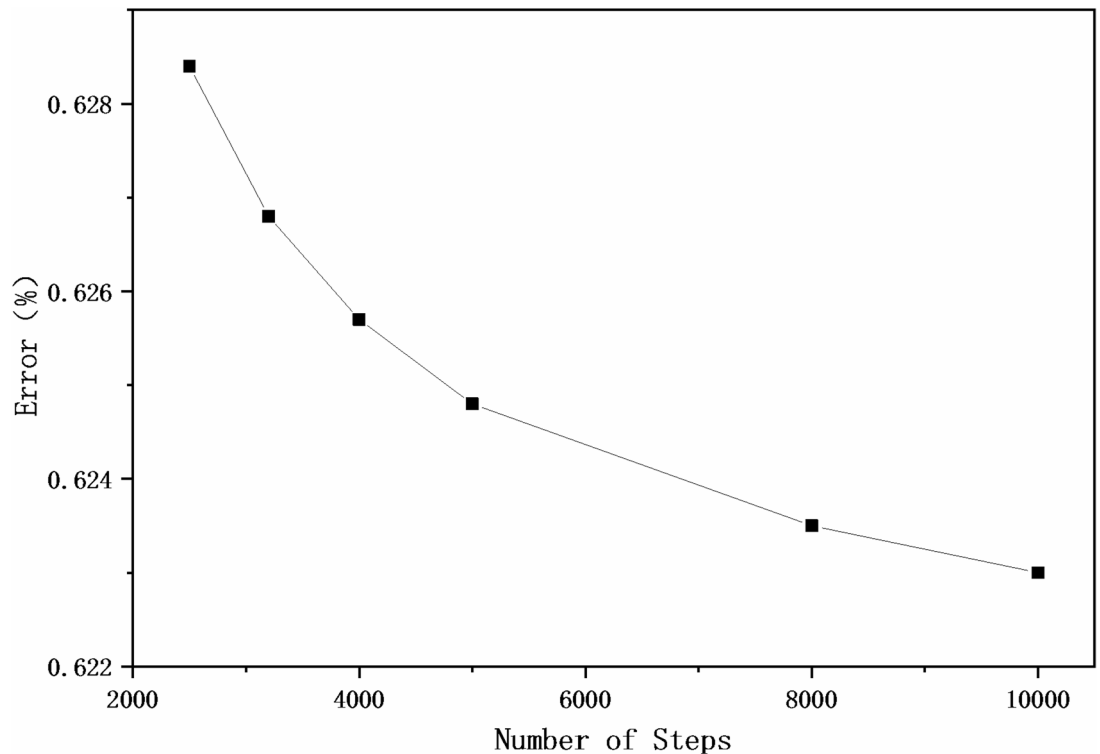


Fig. 5. The convergence of error with the time step is calculated.

Step length	Error
0.008	0.6284%
0.00625	0.6268%
0.005	0.6257%
0.004	0.6248%
0.0025	0.6235%
0.002	0.623%

Table 3. Table of convergence of calculation error with different step lengths.

$$u(x, y, z, t) = 2x^3 + y^3 + 2.5z^3 + 6xt^2 + 3yt^2 + 7.5zt^2 \quad (x, y, z) \in \Omega$$

Five sample points are listed in Table 4.

The considered time interval lasts from 0s to 2s. In this application, the step length is 0.005. The results obtained by two methods are illustrated in Fig. 8. The pressures computed through the RKDRM are stated by discretized symbols while the pressures computed through the PIDRM are stated by curves. In the comparison implementation of PIDRM, a constant boundary is assumed in each time step. In PIDRM, the subdivision parameter is evaluated by 10 and the number of truncated terms is 6 which is the same as that in Zhou's work¹⁹.

As can be seen from Fig. 8, the pressure results that are obtained by the RKDRM coincides with that from the PIDRM.

The accuracy of the Runge-Kutta dual reciprocity boundary element method and that of the PIDRM are further compared through the implementations with different time steps. In the comparison application, 126 boundary nodes, 24 quadratic units and 139 RBF interpolation points are involved in the discretization. Five time steps involving 0.01, 0.005, 0.004, 0.0025 and 0.002 are employed. Figure 9 illustrates the average relative error of pressures among all the sampling points at the end of the variation time.

For a more precise and visual representation, the data in the above chart are listed in the following table.

From Fig. 9; Table 5, it can be seen clearly that the average errors of both methods are small. The accuracy of the RKDRM is higher than that of the PIDRM especially in the case of larger time step length. The errors in the application of both the PIDRM and the RKDRM converge stably with the increasing number of time steps. This comparison indicates that the RKDRM is more suitable for problems with rapidly variable boundary condition.

It is worth noting that the boundary quantities is assumed to be steady in each time step in the PIDRM. Thus, the discretization error is inevitably introduced in this example. We can further improve the accuracy of

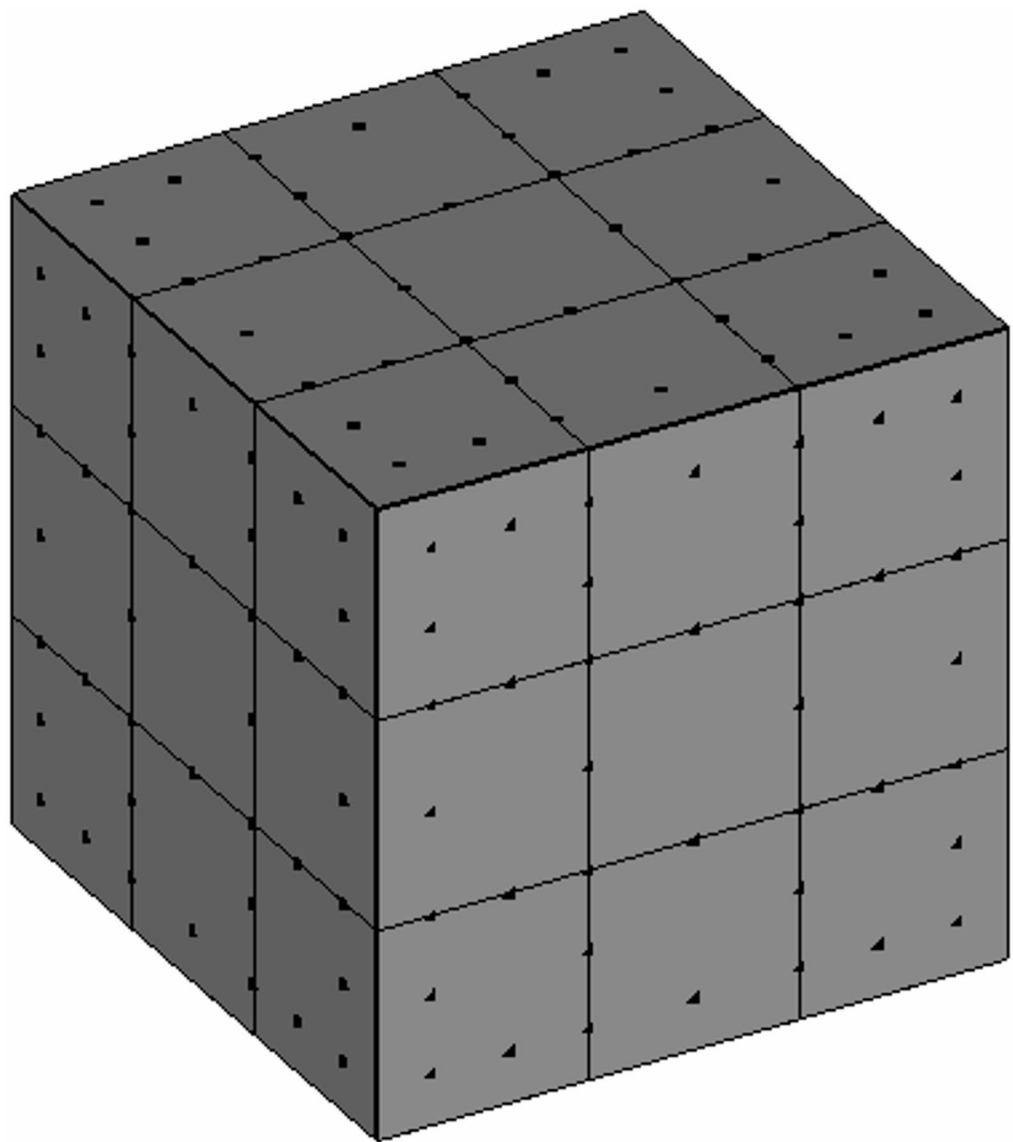


Fig. 6. Boundary element and node distribution.

the PIDRM by utilizing a time convolution integral which is related to the boundary quantities. However, much more computational time should be consumed to accurately compute the convolution.

To further illustrate the efficiency of the presented method, the time consumptions is also compared between the PIDRM and the RKDRM. For the same level of accuracy, the time step 0.005 is considered in the RKDRM application. The time step 0.002 is considered in the PIDRM application. The time cost in both applications are illustrated in Fig. 10. The Intel I5-9400 F CPU with 2.9 GHz frequency.

As can be concluded from Fig. 10, the RKDRM is much faster than the PIDRM to achieve the result at the same accuracy level. This is because in the implementation of PIDRM, a lot of matrix multiplication operations are performed to compute the matrix exponential function in the first step. In the RKDRM, however, less matrix multiplications or inversions are involved.

Conclusion

A coupled DRBEM and Runge–Kutta method has been developed for solving time-domain scalar wave problems. The approach utilizes a steady-state fundamental solution and RBF interpolation to convert the problem into a boundary-only form, leading to a system of second-order ODEs after discretization. These are transformed into first-order form and integrated in time using the fourth-order Runge–Kutta method. Numerical examples demonstrate the method's accuracy, stability, and computational efficiency compared to existing techniques.

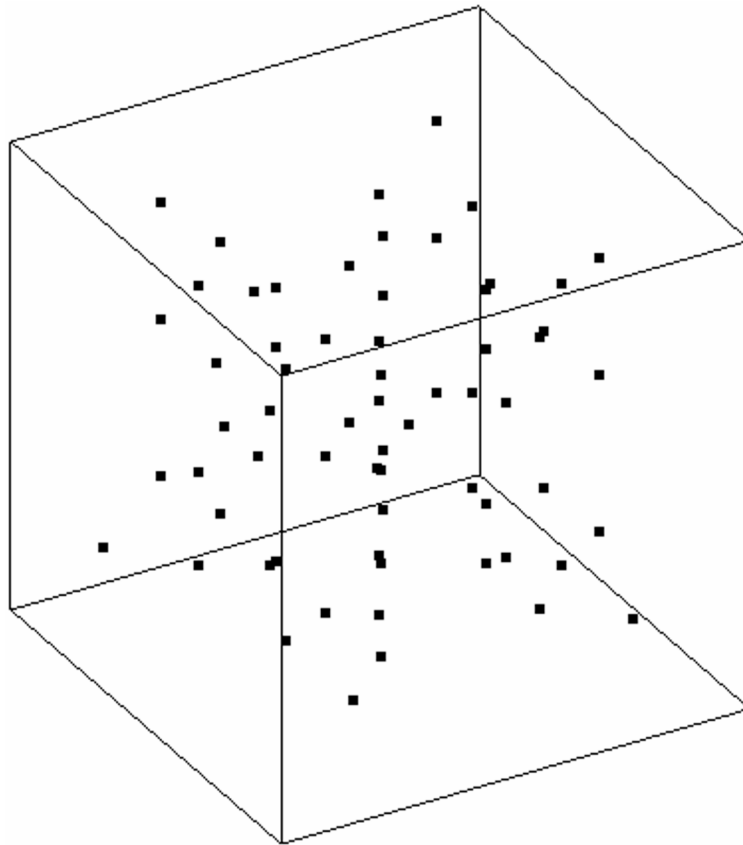


Fig. 7. Distribution of RBF interpolation points in the cube.

Number	Position coordinates (x, y, z)
P1	(0.7917, 0.6, 0.3333)
P2	(0.2083, 0.2, 0.6667)
P3	(0.8, 0.2083, 0.9167)
P4	(0.7917, 0.8, 0.6667)
P5	(0.4, 0.3333, 0.7917)

Table 4. Coordinates of cube sampling points.

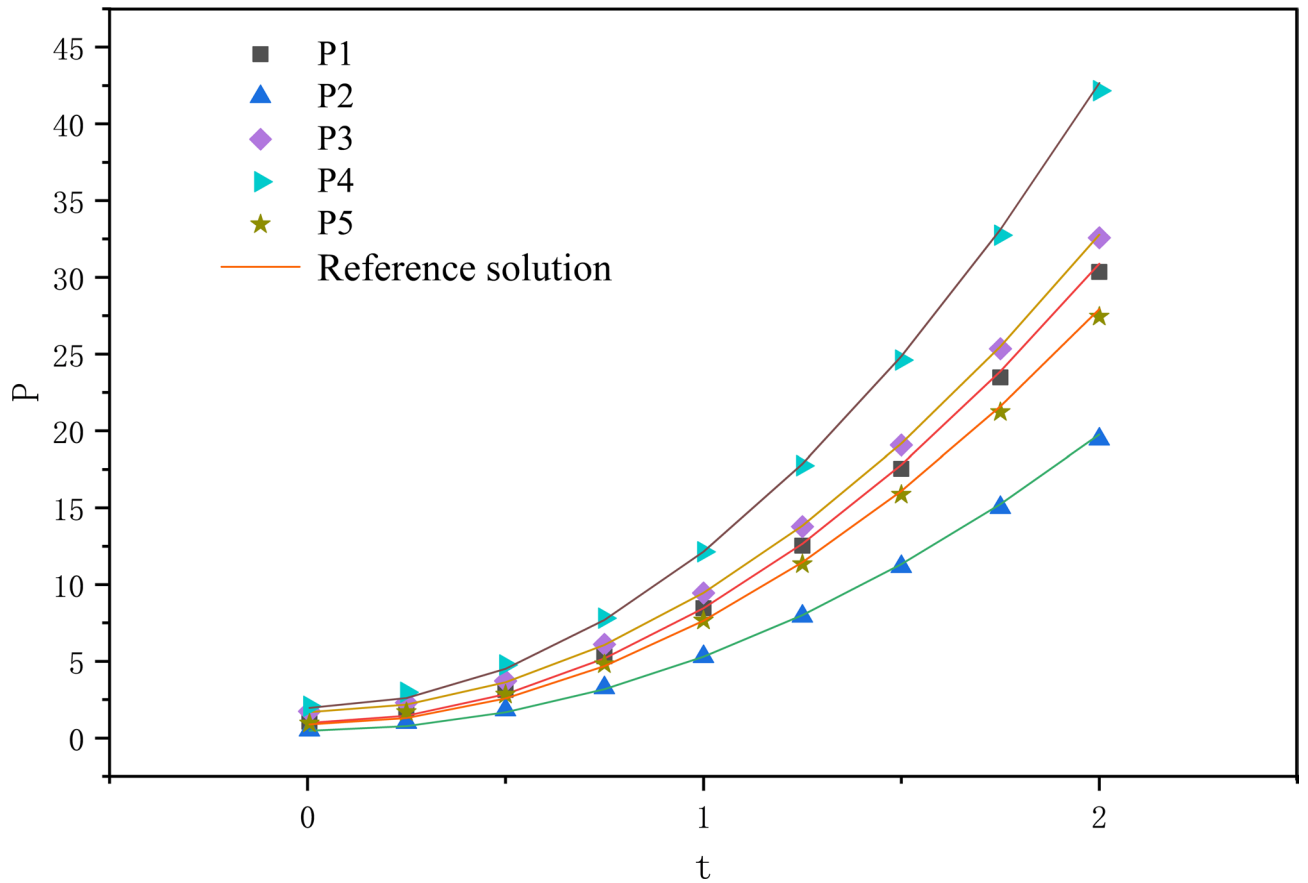


Fig. 8. The calculated value and reference value of the target point sound pressure amplitude.

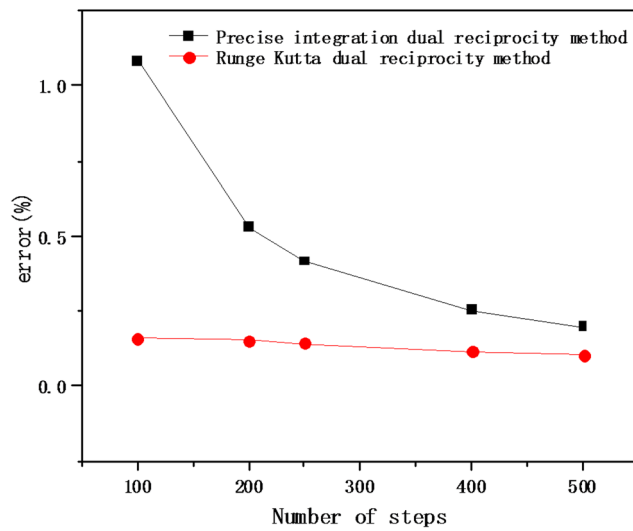


Fig. 9. The relative errors of the two methods.

Step length	Time step	PIDRM	RKDRM
0.01	100	1.083%	0.1609%
0.005	200	0.528%	0.1521%
0.004	250	0.4174%	0.1414%
0.0025	400	0.2526%	0.115%
0.002	500	0.1984%	0.1049%

Table 5. The relative errors with increasing number of time steps.

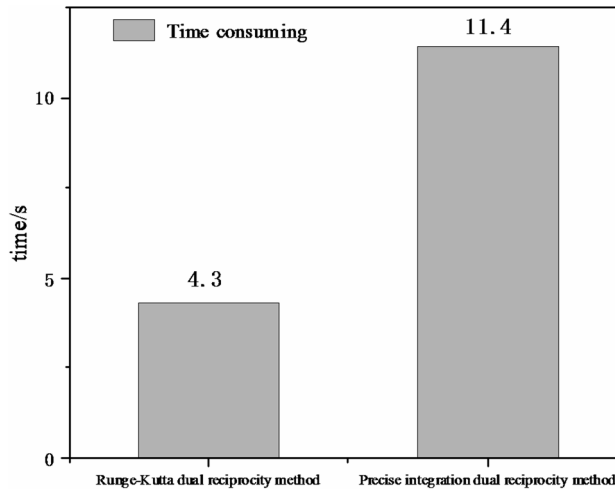


Fig. 10. Comparison of calculation time between two methods with similar errors.

Data availability

The datasets used or analysed during the current study available from the corresponding author on reasonable request.

Received: 4 September 2025; Accepted: 9 December 2025

Published online: 11 December 2025

References

- Rao, S. M. & Raju, P. K. Application of the method of moments to acoustic scattering from multiple bodies of arbitrary shape[J]. *J. Acoust. Soc. Am.* **86** (3), 1143–1148 (1989).
- Liu, X. et al. A fast multipole boundary element method for half-space acoustic problems in a subsonic uniform flow[J]. *Eng. Anal. Boundary Elem.* **137**, 16–28 (2022).
- Kirkup The boundary element method in acoustics: A Survey[J]. *Appl. Sci.* **9** (8), 1642 (2019).
- Ergin, A. A., Shanker, B. & Michielssen, E. The plane-wave time-domain algorithm for the fast analysis of transient wave phenomena[J]. *IEEE Antennas Propag. Mag.* **41** (4), 39–52 (1999).
- Chen, W., Li, J. & Fu, Z. Singular boundary method using time-dependent fundamental solution for scalar wave equations[J]. *Comput. Mech.* **58** (5), 717–730 (2016).
- Belhocine, A. & Ghazaly, N. M. Effects of material properties on generation of brake squeal noise using finite element method[J]. *Latin Am. J. Solids Struct.* **12** (8), 1432–1447 (2015).
- De, S., Rivera, Z. B. & Guida, D. Finite element analysis on squeal-noise in railway applications[J]. *FME Trans.* **46** (1), 93–100 (2018).
- Marklein, R. The Finite Integration Technique as a General Tool To Compute Acoustic, Electromagnetic, Elastodynamic, and Coupled Wave Fields[J].
- Bin, H. et al. A fast multipole boundary element method based on higher order elements for analyzing 2-D elastostatic problems[J]. *Eng. Anal. Boundary Elem.* **130**, 417–428 (2021).
- Von Estorff, O., Pais, A. L. & Kausel, E. Some observations on time domain and frequency domain boundary elements[J]. *Int. J. Numer. Methods Eng.* **29** (4), 785–800 (1990).
- Schneider, S. et al. Validation of the time-domain boundary element method in acoustics regarding flexible multibody simulations and acoustic measurements [J]. *IOP Conf. Ser. Mater. Sci. Eng.* **1190**(1), 012007 (2021).
- Takahashi, T. A fast time-domain boundary element method for three-dimensional electromagnetic scattering problems[J]. *J. Comput. Phys.* **482**, 112053 (2023).
- Frangi, A. Causal shape functions in the time domain boundary element method[J]. *Comput. Mech.* **25**(6), 533–541 (2000).
- Gao, X. W. The radial integration method for evaluation of domain integrals with boundary-only discretization[J]. *Eng. Anal. Boundary Elem.* **26** (10), 905–916 (2002).
- Neves, A. C. & Brebbia, C. A. The multiple reciprocity boundary element method in elasticity: A new approach for transforming domain integrals to the boundary[J]. *Int. J. Numer. Methods Eng.* **31** (4), 709–727 (1991).
- Meral, G. & Tezer-Sezgin, M. DRBEM solution of exterior nonlinear wave problem using FDM and LSM time integrations[J]. *Eng. Anal. Boundary Elem.* **34** (6), 574–580 (2010).

17. AL-Bayati, S. A. & Wrobel, L. C. A novel dual reciprocity boundary element formulation for two-dimensional transient convection–diffusion–reaction problems with variable velocity[J]. *Eng. Anal. Boundary Elem.* **94**, 60–68 (2018).
18. Yu, B. et al. Isogeometric dual reciprocity boundary element method for solving transient heat conduction problems with heat sources[J]. *J. Comput. Appl. Math.* **385**, 113197 (2021).
19. Zhou, F. et al. The precise integration method for semi-discretized equation in the dual reciprocity method to solve three-dimensional transient heat conduction problems[J]. *Eng. Anal. Boundary Elem.* **95**, 160–166 (2018).
20. Oliveira, H. L. & Leonel, E. D. A BEM formulation applied in the mechanical material modelling of viscoelastic cracked structures[J]. *Int. J. Adv. Struct. Eng.* **9** (1), 1–12 (2017).
21. Carrer, J. A. M. & Mansur, W. J. Time-domain BEM analysis for the 2D scalar wave equation: initial conditions contributions to space and time derivatives [J]. *Int. J. Numer. Methods Eng.* **39** (13), 2169–2188 (1996).
22. Zhang, H. et al. Highly efficient invariant-conserving explicit Runge-Kutta schemes for nonlinear hamiltonian differential equations [J]. *J. Comput. Phys.* **418**, 109598 (2020).
23. Melenk, J. M. & Rieder, A. Runge-Kutta Convolution quadrature and FEM-BEM coupling for the time-dependent linear Schrödinger equation[J]. *J. Integr. Equ. Appl.*, **29**(1). (2017).
24. Fan, Y. & Qiu, C. A fourth-order Runge-Kutta method for direct solution of second-order ordinary differential equations[J]. *Int. J. Numer. Methods Eng.* **113** (8), 1218–1232 (2020).
25. Huang, C. S., Lee, C. F. & Cheng, A. H. D. Error estimate, optimal shape factor, and high precision computation of multiquadric collocation method. *Eng. Anal. Bound. Elem.* **31**, 614–623 (2007).

Author contributions

Fenglin Zhou and Jiehua Deng wrote the main manuscript text. Junfeng Man and Yu Qin prepared the illustration figures. Fenglin Zhou, Wenyuan Li, Yijing Zhang and Jianghong Yu coded the program and collected the data.

Funding

This paper is supported partly by the natural science foundation of Hunan Province (2023JJ50163), partly by the Key Laboratory in College of Hunan Province with grant numbers 2023–213 (Xiang Jiao Tong) and partly by Science and Technology Plan Project of Changsha(kzd251101132).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.Z. or W.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025