

Multigrid with FFT smoother for a simplified 2D frictional contact problem

Jing Zhao^{1,*,\dagger}, Edwin A. H. Vollebregt^{1,2} and Cornelis W. Oosterlee^{1,3}

¹*Delft Institute of Applied Mathematics, Delft University of Technology, Delft, The Netherlands*

²*VORtech BV, Delft, The Netherlands*

³*CWI—Center for Mathematics and Computer Science, Amsterdam, The Netherlands*

SUMMARY

This paper aims to develop a fast multigrid (MG) solver for a Fredholm integral equation of the first kind, arising from the 2D elastic frictional contact problem. After discretization on a rectangular contact area, the integral equation gives rise to a linear system with the coefficient matrix being dense, symmetric positive definite and Toeplitz. A so-called fast Fourier transform (FFT) smoother is proposed. This is based on a preconditioner M that approximates the inverse of the original coefficient matrix, and that is determined using the FFT technique. The iterates are then updated by Richardson iteration: adding the current residuals preconditioned with the Toeplitz preconditioner M . The FFT smoother significantly reduces most components of the error but enlarges several smooth components. This causes divergence of the MG method. Two approaches are studied to remedy this feature: subdomain deflation (SD) and row sum modification (RSM). MG with the FFT + RSM smoother appears to be more efficient than using the FFT + SD smoother. Moreover, the FFT + RSM smoother can be applied as an efficient iterative solver itself. The two methods related to RSM also show rapid convergence in a test with a wavy surface, where the Toeplitz structure is lost. Copyright © 2014 John Wiley & Sons, Ltd.

Received 18 May 2013; Revised 25 October 2013; Accepted 9 December 2013

KEY WORDS: integral equation; Toeplitz matrices; multigrid method; fast Fourier transform; subdomain deflation; row sum modification; frictional contact problems

1. INTRODUCTION

A difficult problem in mechanical engineering concerns the elastic contact between two surfaces [1, 2]. Two bodies are pressed together so that a contact area is formed. The question is to find both the region where the bodies are in contact and the pressure that arises between them. Next, when the bodies are brought into relative motion, frictional stress arises between them as well. For large motion, a state of full sliding will occur. But if the relative velocity is small, a creeping motion may occur that is largely affected by the elastic deformations of the two bodies in and around the contact patch. The question is then to find out which parts of the surfaces are sticking together versus where local relative sliding occurs, and further to find the distribution of frictional stress that occurs. The results of this are used to investigate diverse aspects such as friction and wear [3, 4], rolling contact fatigue [5], and the fatigue life of machine elements [6].

The frictional contact problem has been studied by many in the past decades, where the works of Johnson [1] and Kalker [2] are regarded as fundamental. For a more recent overview of the physical phenomena and mathematical modeling, we refer to [7]. Its complexity derives from Coulomb's

*Correspondence to: Jing Zhao, Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD, Delft, The Netherlands.

[†]E-mail: J.zhao-1@tudelft.nl

friction law, which is applied locally in each point of the contact patch. This gives a bound for the tangential stress on the contacting surfaces, formed as the product of normal pressure and a friction coefficient. Local slip occurs where the magnitude of tangential stress reaches the traction bound. There, the tangential stress takes the direction opposite to the local slip velocity. Coulomb's law itself involves an inequality constraint. Furthermore, this constraint is nonlinear in 3D problems. As a result, the solution of frictional contact problems is quite complicated and takes considerable computing time, so that faster solution methods need to be sought.

Generally, two numerical methods have been employed to solve frictional contact problems. The finite element method [8–10] is widely used, especially for the problems related to nonlinear materials. However, this method is computationally expensive because of the full discretization of the whole contacting bodies. The boundary element method (BEM), for example, [11, 12], is efficient for solving homogeneous elastic problems. This method transforms a boundary value problem into a boundary integral equation and then solves it only at the boundary, which is discretized. In this way, the computational work is greatly reduced.

The BEM works particularly well for so-called concentrated contact problems, where the contact patch covers only a small portion of the contacting bodies' boundaries. The dominant method for solving rolling contact with friction is Kalker's variational approach [2], which is implemented in the CONTACT software [13]. This is a BEM that uses Green's functions for the elastic half-space. Further, it uses the normal (NORM) and tangential (TANG) algorithms for identifying the contact area and its subdivision into the parts where sticking or local sliding occurs. These are active set algorithms that deal with the inequality constraints. In each active set iteration, a contact area is fixed, and the governing equations are solved. The resulting tractions are used to determine a new contact area according to the contact conditions. Vollebregt developed an iterative solution method that is now used as an alternative to the TANG algorithm. This method is of block Gauss–Seidel type and is called ConvexGS [14].

Solving the surface integral problem has always been the most time-consuming part. Within the context of a prescribed contact area, this integral equation is a Fredholm integral of the first kind. Using rectangular elements for discretization on a rectangular domain, the normal contact problem leads to a linear system with its coefficient matrix being dense, symmetric and positive definite (SPD). The 2D tangential contact problem is also linear, but in 3D, the tangential problem becomes nonlinear. Moreover, the underlying matrix is a Toeplitz matrix for 2D problems and a block Toeplitz matrix with Toeplitz blocks (BTTB) for 3D problems. This structure can be exploited using fast algorithms.

Toeplitz matrices occur in various engineering fields, for example, image deblurring and signal processing (e.g., [15, 16]). Besides the direct methods, for example, [17], a popular iterative method for these matrices is the preconditioned conjugate gradient algorithm, where circulant and Toeplitz preconditioners were studied in [18–21]. As an efficient iterative method, multigrid (MG) has also been applied and analyzed, for example, in [22–25]. From the point of computational cost, Lubrecht and Venner [26] applied a multi-level multi-summation technique. Finally, fast Fourier transforms (FFTs) have been used successfully for contact problems too [27], for speeding up matrix–vector multiplications for Toeplitz matrices.

The motivation of our work is to improve the computational speed for solving frictional contact problems. Previously, we developed a full MG method for the normal contact problem [28]. An important component in this involved the restrictions and interpolations of functions on changing, irregular domains. At that time, we used Gauss–Seidel iteration as a smoother, because of its robust convergence properties. However, this smoother leads to $\mathcal{O}(n^2)$ complexity for the overall solution algorithm. In a separate work, we addressed the use of FFTs for speeding up the NORM algorithm and found a way to use FFTs for creating a preconditioner as well [29]. Here, we investigate the possibilities of this algorithm as a smoother in the MG algorithm. This appears to be not so straightforward as one might hope. Therefore, this paper focuses on a 2D problem in which no slip occurs, and aims to clearly expose the difficulties and the remedies that can be used.

This paper consists of six sections. Section 2 formulates the 2D frictional rolling contact problem. Furthermore, it specifies the first test problem for later discussions on the efficiency and convergence. Our FFT smoother is proposed in Section 3, including the construction and discussion of its

smoothing properties. The investigation of the FFT smoother in the MG method shows its ability to significantly reduce most of the components of error except for several problematic modes. Section 4 studies two different remedies: subdomain deflation (SD) and row sum modification (RSM). Three numerical tests are implemented in Section 5. In addition, the FFT + RSM as a solver, rather than as a smoother, is also discussed. The last section concludes the paper.

2. FORMULATION OF THE 2D FRICTIONAL CONTACT PROBLEM

This section gives a general introduction of the 2D frictional contact problem along the lines of the presentation in [30]. More details can be found in [2]. The first test case, called ‘test 1’, is defined at the end of this section. It will be used in the investigation and discussion in the sections to follow.

2.1. Physics of the contact problem

In order to give a physical description of contact phenomenon, we take railway contact as an example. The overall geometry, drawn in Figure 1(a), displays a wheel with radius R rolling on the rail with a forward velocity V and angular velocity ω . The rail is bent a bit because of the load of the wheel, F_w , and from the sleepers, F_{s1} , F_{s2} . Beyond this initial deformation, both the wheel and the rail are considered rigid in this picture. The wheel is shifted down so that some overlap (penetration) with the rail occurs. This may be seen more clearly in Figure 1(b) when we stretched the graph vertically. This is called the ‘undeformed state’.

In the simulation of the dynamic behavior in such a system, one searches for the position and the speed of the wheel, and the bending of the rail due to the influence of all forces. The forces can be found to be a function of the profile of the contacting bodies and the maximum penetration. These give elastic deformation, which is shown in Figure 1(c) as the ‘deformed state’. After deformation, the surfaces of the two bodies coincide, and an area of contact occurs.

Hence, with the position information in the undeformed state, we can solve for the resulting deformation, including the contact area and the traction on it. This traction includes pressure in normal direction and frictional stress acting tangentially. The former could be obtained by the classical Hertzian theory [31] if the contacting surface is smooth and quadratic or with a numerical theory (e.g., [13]) if the situation is non-Hertzian. It is compressive in the contact area and zero outside. The latter results from ‘creepage’. This is the average relative velocity between the surfaces and is given by the difference between the forward velocity V and the circumferential velocity ωR . Creepage is accommodated partly by elastic deformation and partly by local sliding in the contact patch. In both cases, frictional stress is invoked, to create deformation for either keeping the surfaces sticking together or opposing the local slip velocity.

A typical size for the contact between steel wheel and rails is 10×10 mm, and the corresponding penetration is 0.01 mm. The elastic deformations decrease with $1/r$, with r the distance to the contact area, and the stresses and strains decrease in proportion to $1/r^2$. Therefore, the stresses and strains are negligible at distances of a few centimeters outside the contact patch. This is the reason why the overall and local problems may be decoupled [2].

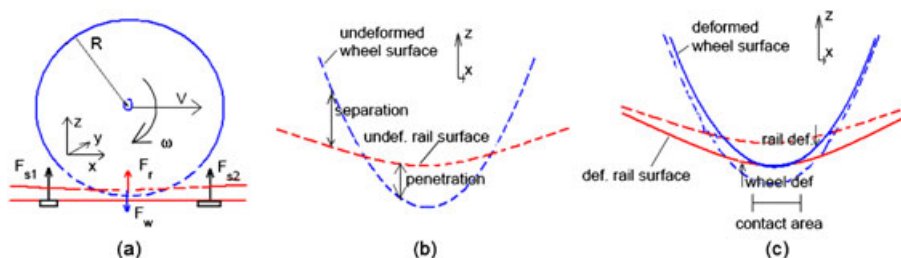


Figure 1. Geometry of contact problems for railway applications [30]: (a) the overall geometry, (b) undeformed state, and (c) deformed state.

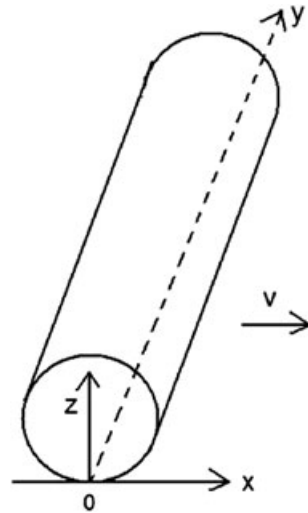


Figure 2. The 2D line contact problem.

2.2. The mathematical model of frictional contact

For a detailed study of the components of the contact solver, this paper focuses on a simple 2D frictional contact problem. Figure 2 depicts a cylinder rolling over a flat surface, with its axle parallel to the y -direction. This is a situation of ‘line contact’ where ‘plane strain’ occurs. In this situation, there is no dependence on the y -coordinate, so that we go from a 3D to 2D situation.

Modeling starts from the assumptions of linear elasticity, small displacements and displacement gradients, and uses a quasi-static approach. The basic equations obtained from these assumptions describe equilibrium of the forces everywhere in the contacting bodies at all times (ignoring effects of inertia), give the relation between displacements and strains, and describe the strain of the material as a function of the stresses applied (constitutive equation). We are interested mostly in the quantities at the surfaces of contacting bodies. As the tractions on the two bodies are equal and opposite, we need only consider the tractions acting on one of them. Moreover, if the bodies are of equal elastic materials, the displacements are of equal size and opposite too, and can be combined into a single ‘displacement difference’, called displacement for short.[‡]

The 3D tractions acting on the upper surface are denoted by \mathbf{p} . The component in normal direction is $p_n = \mathbf{p} \cdot \mathbf{n}$ with $\mathbf{n} = (0, 0, -1)^T$ the outer normal (coordinate system aligned with the contact plane). The pressure p_n is compressive in the contact area C and vanishes outside. Assuming that the normal contact problem has been solved before, p_n and C are prescribed, we concentrate on the tangential problem. Further, as we concentrate on a 2D situation, the unknowns in the contact plane are functions of the position x and the time instant t .

Three aspects, that is, creepage, elastic deformation and friction between two contacting bodies are taken into account in the model. First of all, the rigid slip velocity w_x in the undeformed state must add up with the actual slip velocity s_x and the change of deformation Du_x/Dt [30]:

$$s_x(x, t) = w_x(x, t) + \frac{Du_x(x, t)}{Dt}. \quad (1)$$

A material time derivative is used for the change of deformation so that the same equation can be used with moving coordinates. In case of wheel-rail contact with a world-fixed coordinate system, this term equals $\partial u_x / \partial t$.

[‡]The situation of equal elastic constants is called quasi-identity [2] and is assumed here for simplicity; the same theory works for bodies of different materials as well.

Concerning the third aspect, friction, the contact area C is divided into an adhesion and a slip area, determined by the following contact conditions, amounting to Coulomb's friction law applied locally in each point $x \in C(t)$:

$$\text{In adhesion area } H(t): |s_x(x, t)| = 0, \quad |p_x(x, t)| \leq g(x, t); \quad (2)$$

$$\text{In slip area } S(t): |s_x(x, t)| > 0, \quad p_x(x, t) = -g(x, t) \operatorname{sgn}(s_x(x, t)). \quad (3)$$

These conditions state that the tangential traction is bounded by a traction bound $g(x, t) = \mu p_n(x, t)$, where μ is the friction coefficient. When the tangential traction reaches the maximum, local slip occurs, and the traction opposes the slip. In 2D, this simply requires to find the sign of the slip velocity, which is typically the same as the sign of w_x . In 3D, this problem becomes much more complicated. Equation (3) can then be formulated as $p_x^2 + p_y^2 = g^2$ and $p_x \cdot s_y - p_y \cdot s_x = 0$, which is nonlinear.

The second aspect mentioned earlier, concerning the deformation as function of stresses p_x , is addressed subsequently.

2.3. The half-space approach

The slip velocity s_x in (1) depends on the overall motion (the rigid slip velocity w_x) and the rate of change of the deformation Du_x/Dt . The method to calculate the deformation $u_x(x, t)$ is based on four simplifying assumptions in Kalker's variational approach [2]. First of all, the contacting bodies are made of homogeneous linear elastic materials. Second, the contact area is small compared to the size of the contacting bodies themselves and hence is considered to be *flat*. Moreover, there are no sharp variations in the geometry. The fourth assumption is to ignore the effects of inertia on the motion of surface particles.

These assumptions allow for the use of the *half-space approach*. The idea of this is to approximate the elastic field in the contacting bodies, by considering each body as a half-space, a semi-infinite elastic solid bounded by a plane surface. Based on the classical solutions by Boussinesq and Cerruti (see Johnson [1] or Kalker [2]), the relation between surface tractions and deformations is the following:

$$u_x(x, t) = \int_{C(t)} A(x, x') p_x(x', t) dx'. \quad (4)$$

This integral indicates that at one time instant, the deformation u_x at one point x is influenced by the tractions in the whole contact area. Note that relation (4) is instantaneous, there is no dependence on the prior history. This is because of the quasi-static approach in which the effects of inertia on the elastic field are ignored. The kernel function $A(x, x')$ is the influence function for deformation at the surface point x , because of the contribution of a unit traction at another surface point x' . We can write

$$A(x, x') = A(x - x'), \quad (5)$$

to indicate that the function depends on the relative position of the two points, rather than on their absolute positions. This property leads to a special structured matrix after discretization in the following.

2.4. Discretization

In the 2D problem, shown in Figure 2, the contact area is a strip of infinite length in the y -direction. Hence, the real contact area is $[-a, a] \times [-\infty, \infty]$. One way of dealing with this infinity is by truncating the contact area to $[-a, a] \times [-b, b]$ with $a \ll b$. We only use a single element for discretization in the y -direction and n elements in the x -direction. Quantities in y -direction can then again be ignored because of plane strain. An advantage of this approach is that 2D problems can be solved with the same code as 3D problems, without any reference to the influence functions for 2D.

Now, we discuss the discretization of the integral (4) and the slip equation (1) on a rectangular domain. A cell-centered mesh is used. The surface traction is approximated by an element-wise constant function. Then the integral (4) is discretized as

$$u_I = \sum_{J=1}^n A_{I,J} p_J, \quad I = 1, \dots, n, \tag{6}$$

where I and J are element indices. In a matrix form, we obtain

$$\mathbf{u} = A \mathbf{p}, \quad \mathbf{u}, \mathbf{p} \in \mathbb{R}^n, \quad A \in \mathbb{R}^{n \times n}. \tag{7}$$

The coefficient matrix A is dense and SPD. In 2D, it has a Toeplitz structure because of the space-invariance property in (5), and it becomes ill conditioned when the problem size gets large. In 3D problems, the coefficient matrix is BTTB.

In our MG approach for the 3D normal contact problem [28], we used a Gauss–Seidel smoother. Because A is dense, this smoother has arithmetic complexity $\mathcal{O}(n^2)$ and is difficult to parallelize. This solver is therefore not competitive to other algorithms based on FFTs (exploiting the Toeplitz structure) or multi-level summation (also exploiting (5)) that are of complexity $\mathcal{O}(n \log(n))$. Therefore, our target is to construct a smoother that is parallelizable and has $\mathcal{O}(n \log(n))$ complexity, for solving the tangential contact problem.

The slip equation (1) in rolling problems can be discretized with a sequence of time steps with step length $\delta t = t - t'$, where t and t' are the current and previous time instants, respectively. Then Equation (1) yields [13]

$$\mathbf{s} = \mathbf{w} + (\mathbf{u} - \mathbf{u}')/\delta t. \tag{8}$$

This uses the implicit Euler approach.

2.5. The first test

We specify a test problem, test 1, for the following discussion about efficiency and convergence of the smoother in our MG solver. It is defined as follows:

- The cylinder in Figure 2 is pressed first to a flat surface and then shifted, instead of rolled. Hence, the deformation at a previous time step \mathbf{u}' is zero, and the time dependence is not present.
- We assume that the friction coefficient $\mu = \infty$. In this case, the traction will not reach the traction bound and no slip occurs. Equation (8) becomes

$$\begin{aligned} \mathbf{s} &= \mathbf{w} + \mathbf{u}/\delta t, && \text{(since } \mathbf{u}' = \mathbf{0}) \\ &= \mathbf{w} + A\mathbf{p}/\delta t && \text{(due to Equation (7))} \\ &= \mathbf{0} && \text{(due to } \mu = \infty). \end{aligned} \tag{9}$$

Therefore, the system to be solved is

$$A\mathbf{p} = \mathbf{u}, \tag{10}$$

where $\mathbf{u} = -\delta t \mathbf{w}$ with $(\delta t \mathbf{w})$ the rigid slip.

- The cylinder and the flat surface have shear modulus $G = 82,000 \text{ N/mm}^2$ and Poisson ratio $\nu = 0.28$, typical values for steel. These two parameters are used to compute the influence coefficients.
- The potential contact area is $[-4, 4] \times [-50, 50] \text{ mm}^2$. We use an $n \times 1$ grid for discretization, yielding a Toeplitz coefficient matrix A in Equation (10).
- We set the rigid shift $(\delta t \mathbf{w})$ to be constant with 0.0008 mm , and we solve for the traction \mathbf{p} .

Summarizing, test 1 reduces the commonly 3D problem to a 2D problem. For simplicity, it excludes time dependence and contact conditions. We solve Equation (10) to obtain the tractions \mathbf{p} . The right-hand side of this system is constant in test 1, but we will also consider a space-dependent right-hand side in test 2 in Section 5.

3. FFT SMOOTHER FOR MULTIGRID METHOD

The MG method shows its great efficiency particularly when solving elliptic PDEs. It has also been developed for integral equations of the first kind (e.g., [32, 33]) and of the second kind (e.g., [34–36]). The integral (4) in the 2D problem is the first kind. As a smoother is essential for an MG solver, we focus on finding a satisfactory smoother in this paper. Our target is to design a solver of $\mathcal{O}(n \log(n))$ complexity with n the number of unknowns.

This section consists of three subsections. We construct the FFT smoother in the first subsection and analyze its smoothing properties in the second subsection. The last subsection shows MG convergence with this smoother.

3.1. Construction of the FFT smoother

Consider solving a system $A\mathbf{p} = \mathbf{u}$, where \mathbf{p} and \mathbf{u} are the tractions and deformations, respectively. The FFT smoother is inspired by the idea of using FFTs for matrix–vector products (MVPs) for Toeplitz matrices. It is based on a Richardson iteration without relaxation parameter, which reads as follows:

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \hat{\mathbf{r}}^k, \quad (11)$$

where

$$\hat{\mathbf{r}}^k = M\mathbf{r}^k, \quad (12)$$

that is, the correction is based on the current residual $\mathbf{r}^k = \mathbf{u}^k - A\mathbf{p}^k$, preconditioned by M . The challenge is to construct the preconditioner M . As M is required to be an accurate approximation for A^{-1} , we analyze the columns in A^{-1} first. From the mechanics point of view, the K th column of A^{-1} represents the traction pattern, which results in a unit deformation on element K and zero deformation on the other elements.

Consider a 16×1 problem in test 1 as an example. Figure 3(a) depicts columns \mathbf{c}^K of A^{-1} , where $K = 1, 3, 8, 12$. The pattern in the K th column reveals the largest positive component at element K and negative components for other elements. Such a pattern physically implies that on element K , a traction is arranged to obtain a unit deformation, with opposite tractions on the other elements to prevent the deformation there.

This figure also shows that, except for the column at the boundary, the other columns of A^{-1} have similar shapes. This implicitly is an indication for the Toeplitz-like structure of matrix A^{-1} . Therefore, we construct a Toeplitz preconditioner M , using the coefficients obtained by approximating only one interior column of matrix A^{-1} .

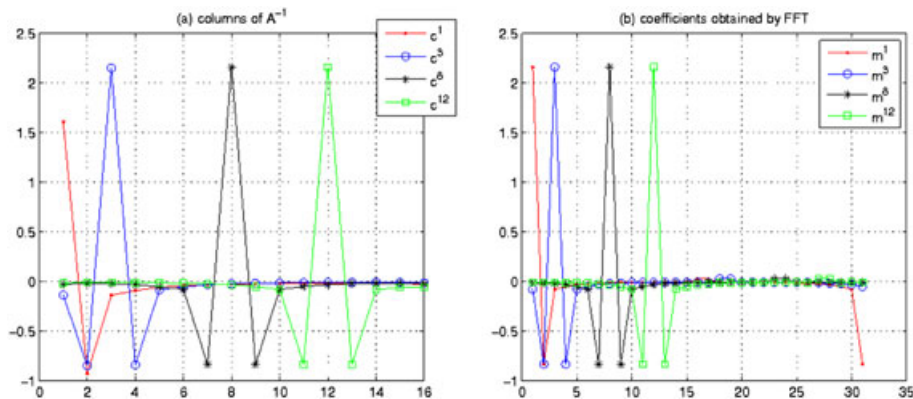


Figure 3. (a) The first, third, eighth, and 12th columns of A^{-1} , and (b) the coefficients obtained by FFTs to approximate them (for 16×1 problem).

- 1. Denote the vector \mathbf{a} with all the influence coefficients, i.e. $\mathbf{a} = [a_{-(n-1)}, \dots, a_{-1}, a_0, a_1, \dots, a_{n-1}]^T$, which uniquely determines this Toeplitz coefficient matrix A .
- 2. Extend the vector $\mathbf{u} = [0, \mathbf{e}]^T$ of the same size with \mathbf{a} , where $\mathbf{e} = [e_0, e_1, \dots, e_{n-1}]$ is a unit vector of size n and the K th component $e_K = 1$.
- 3. The coefficients to approximate the K th column could be obtained by:

$$\mathbf{m}^K = \text{IFFT}(\text{FFT}(\mathbf{u}) / \text{FFT}(\mathbf{a})).$$
 This formula means that both \mathbf{u} and \mathbf{a} are transformed by the FFTs into Fourier domain, where pointwise division (\cdot) is done, then the result is transformed back by the IFFT.

Figure 4. The algorithm to approximate the K th column of A^{-1} by the FFTs in our test problem.



Figure 5. Steps 1 and 2 in the algorithm of approximation by FFTs.

The approximation of such a column can be performed by the FFT technique. It is a powerful technique for fast computing with specially structured matrices, for example, circulant and Toeplitz matrices. Both FFT and inverse FFT reduce the cost of an MVP $A\mathbf{x} = \mathbf{b}$, from $\mathcal{O}(n^2)$ to $\mathcal{O}(n \log(n))$ operations with n the number of unknowns. The main idea is to transform the coefficients of matrix A and vector \mathbf{x} by FFTs to the Fourier domain where point-wise multiplication is implemented, and then we transform back by the inverse FFT to the physical domain. We refer to [37, 38] for more details. Now, if \mathbf{b} is given, \mathbf{x} is approximated by transforming \mathbf{b} and A to the Fourier domain and then by back-transformation after point-wise division.

By this approach, we approximate the K th column of A^{-1} in our test problem. The algorithm is given in Figure 4 with the steps 1 and 2 illustrated in Figure 5. Figure 3(b) shows the resulting approximations for the columns in (a) by this algorithm. As can be seen, the left-half part of the patterns in (b) shows similar shapes as those in (a).

We choose to approximate the center column $K = n/2$ of A^{-1} in our method. The left-half part of the resulting coefficients are denoted by $\mathbf{m} = [m_{-(n-1)}, \dots, m_{-1}, m_0, m_1, \dots, m_{(n-1)}]$, with $m_i = m_{-i}, i = 1, \dots, (n-1)$ and m_0 the largest positive component. The dense symmetric Toeplitz preconditioner M is defined as follows:

$$M = \begin{pmatrix} m_0 & m_{-1} & \cdots & m_{-(n-2)} & m_{-(n-1)} \\ m_1 & m_0 & m_{-1} & \cdots & m_{-(n-2)} \\ \vdots & m_1 & m_0 & \ddots & \vdots \\ m_{n-2} & \cdots & \ddots & \ddots & m_{-1} \\ m_{n-1} & m_{n-2} & \cdots & m_1 & m_0 \end{pmatrix}.$$

3.2. Discussion about smoothing analysis

To analyze the performance of the FFT smoother, we decompose the error \mathbf{e} by the eigenvectors of iteration matrix $R_F := (I - MA)$. A smoother for an MG method should be able to efficiently eliminate the oscillatory eigenvectors. The smooth parts that are left can be represented and resolved on a coarse grid. Figure 6(a) displays the eigenvalues of iteration matrix R_F of the FFT smoother for a 32×1 grid. As can be seen, all eigenvalues are very close to zero, except for two isolated eigenvalues that yield a spectral radius larger than 1. The eigenvectors corresponding to these two problematic eigenvalues are plotted in Figure 6(b). They show either symmetry or anti-symmetry. Significant jumps are found near the boundaries, as the boundary effect is ignored in preconditioner

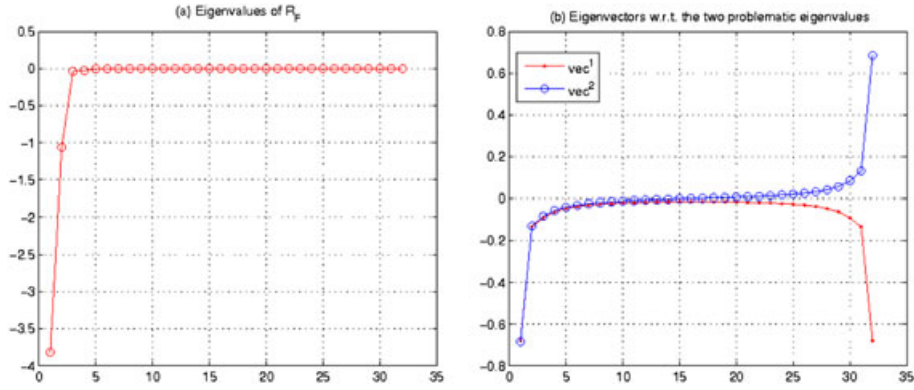


Figure 6. (a) Eigenvalues of $R_F = (I - MA)$ of the FFT smoother on a 32×1 grid. (b) Two eigenvectors corresponding to the problematic eigenvalues.

M but represented in A^{-1} . This observation indicates that the FFT smoother is able to significantly reduce most eigenvectors. However, the two eigenvectors that correspond to the isolated eigenvalues are magnified.

Local Fourier analysis is widely used for smoothing analysis with PDE problems [39], as the Fourier modes are often the same as the eigenvectors of the iteration matrix of the smoother. However, the Fourier modes do not resemble the eigenvectors in our problem! For instance, if we define the error in test 1 in a discretized form using an anti-symmetric extension [40],

$$\mathbf{e} = \sum_{k=1}^{n-1} b_k \sin\left(\frac{k\pi \mathbf{x}}{L}\right),$$

where \mathbf{x} represents the centers of all elements, and $2L$ is the period of the extended error. Fourier modes are given by

$$\mathbf{v}_k = \sin\left(\frac{k\pi \mathbf{x}}{L}\right).$$

Figure 7 shows a low-frequency mode \mathbf{v}_2 in (a) and a high-frequency mode \mathbf{v}_{30} in (b). These two modes are denoted by red dots, and their shapes after the FFT smoother are given by blue circles. Both modes after smoothing display the shape of the anti-symmetry eigenvector, shown in Figure 6(b). (This is due to the Fourier modes we choose. They are sine functions with anti-symmetry shapes. The decomposition of the modes by the eigenvectors has a large portion of the

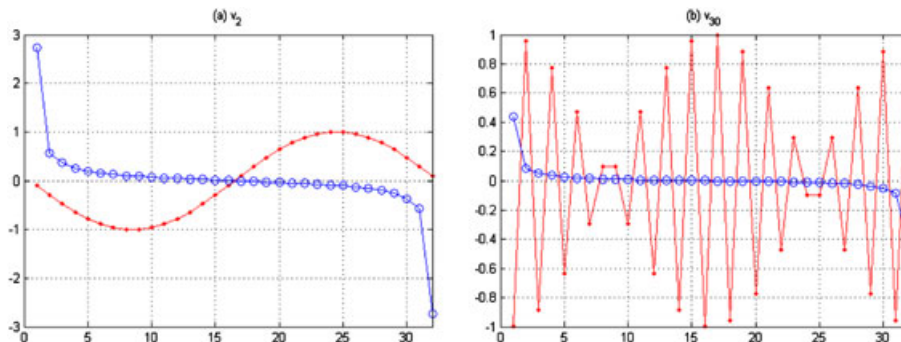


Figure 7. The Fourier modes (in red dots) and after the FFT smoother (in blue circles) for 32×1 grid: (a) a smooth mode \mathbf{v}_2 and (b) an oscillatory mode \mathbf{v}_{30} .

anti-symmetry eigenvector.) It is clear that the Fourier modes are not eigenfunctions. Therefore, we analyze the eigenvectors instead of the Fourier modes in our integral problem.

3.3. The multigrid method based on the FFT smoother

The preceding discussion indicates a difficulty of the FFT smoother when dealing with the two eigenvectors in Figure 6(b). To see how this influences the MG convergence, we specify an MG method with basic coarse grid correction. Standard coarsening, which doubles the element size, is illustrated in Figure 8: two elements on the fine grid are combined into one coarse element.

The components of a two-grid method are defined as follows. It is easily extended to an MG method. The subscripts h and H denote the fine and coarse grid, respectively.

- Smoother: the FFT smoother, as explained.
- Restriction: use the average of defects on the two grid elements on the fine grid for the corresponding coarse grid elements. Taking the green elements in Figure 8 as an example, the restriction reads as follows:

$$d_H^1 = \frac{1}{2} (d_h^1 + d_h^2). \tag{13}$$

- On the coarse grid: the coefficient matrix is obtained according to the grid resolution, and a direct method is used to solve the defect equation. (We employ the backslash command in Matlab.)
- Interpolation: copy the correction from the coarse grid element to the two corresponding fine grid elements. For instance, in Figure 8,

$$v_h^1 = v_h^2 = v_H^1. \tag{14}$$

As MG is an iterative method, it converges when the spectral radius of the iteration matrix is less than 1. In our problem, use 2×1 grid as the coarsest grid. The spectral radii of the MG operators with the FFT smoother are given in Table I, where $V(1,0)$ - and $V(0,1)$ -cycles display the same results because of the symmetry of matrices A and M . As can be seen, the spectral radii of these three V -cycles grow as the problem size increases. And they obtain larger than 1 for larger problems. The $V(1,1)$ -cycle exhibits a much worse spectral radius than other two V -cycles. This is an indication of the difficulty of the FFT smoother. The reason may be explained as follows. The problematic eigenvectors contain steep gradients near the boundaries. This non-smoothness is amplified by the FFT smoother, and the coarse grid correction, with restriction using averaging and interpolation just copying them.

Although the FFT smoother results in MG divergence, it possesses the advantages of easy construction, fast computation of MVPs, and great reduction of major parts of the error. In the next section, we will discuss two remedies to overcome the difficulty with the problematic eigenvectors in Figure 6(b).

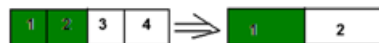


Figure 8. The coarsening approach for a ‘2D problem’. Take the green elements on a 4×1 grid as an example. Two elements on the fine grid are combined to be one coarse grid element.

Table I. The spectral radii of MG operators with the FFT smoother for different discretizations.

Multigrid cycle	8×1	16×1	32×1	64×1	128×1	256×1
$V(1,1)$	1.80e0	1.74e1	2.49e2	5.06e3	1.41e5	5.20e6
$V(1,0)$	2.20e-1	7.1e-1	2.60e0	1.18e1	6.19e1	3.75e2
$V(0,1)$	2.20e-1	7.1e-1	2.60e0	1.18e1	6.19e1	3.75e2

4. REMEDIES FOR THE FFT SMOOTHER

This section describes two attempts to make MG with the FFT smoother convergent, SD and RSM.

4.1. Subdomain deflation

The first technique is to apply the FFT smoother to a residual, which does not contain the problematic eigenvectors of R_F , that is, of $\bar{A} := MA$. In order to remove them, we make use of the *deflation* technique, which preferably leaves the remaining part of the spectrum unchanged [41, 42]. The deflation matrix $P \in \mathbb{R}^{n \times n}$ is defined as

$$P := I - \bar{A}Q, \quad \text{with} \tag{15}$$

$$Q := ZE^{-1}Z^T, \quad \text{and} \tag{16}$$

$$E := Z^T \bar{A}Z. \tag{17}$$

We need to determine a deflation-subspace matrix Z , the columns of which should preferably be approximations of the ‘bad’ eigenvectors. Moreover, computing with the deflation matrix should be cheap. One easy and often used approach is SD [43]. It divides the computational domain Ω into m non-overlapping subdomains Ω_j , $j = 1, 2, \dots, m$, and we presume one deflation vector corresponding to one subdomain. This results in a deflation-subspace matrix Z with m columns. Using a piecewise constant approximation, the entries of Z are defined as

$$z_{ij} = \begin{cases} 1 & i \in \Omega_j \\ 0 & \text{otherwise} \end{cases},$$

where $i = 1, \dots, n$ denote the grid elements, and $j = 1, \dots, m$ are the indices of the subdomains.

Figure 6 showed only two problematic eigenvalues. Hence, we divide the computational domain into $m = 2$ identical subdomains, that is, left-half part and right-half part. For instance, there are $n = 8$ elements in total, with elements $\{1, 2, 3, 4\}$ and $\{5, 6, 7, 8\}$ in the left and right subdomains, respectively. Then the resulting matrix Z reads as follows:

$$Z = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}^T.$$

Figure 9 shows the eigenvalues of the iteration matrix based on this approach, $R_D := (I - PMA)$, for the 32×1 problem. They are denoted by blue circles. Comparing to the eigenvalues of the

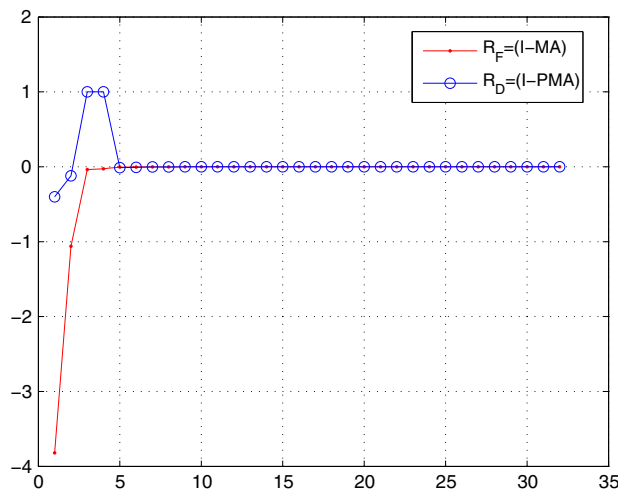


Figure 9. Eigenvalues of R_F with the FFT smoother (denoted by red dots) and of R_D with the FFT + SD smoother (given by blue circles) for the 32×1 problem.

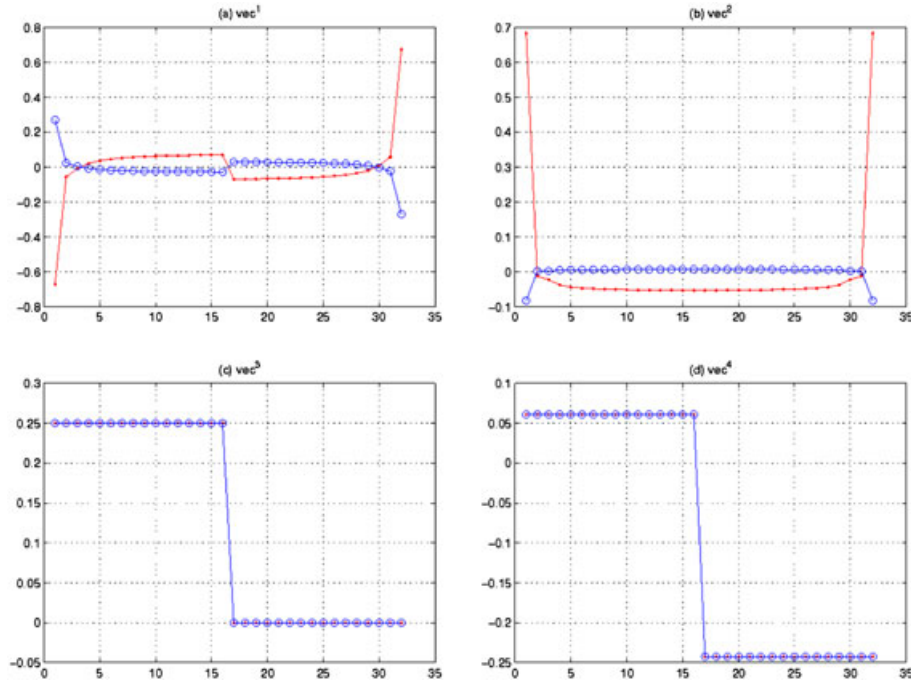


Figure 10. The FFT + SD smoother: The first four eigenvectors of iterative matrix R_D (red dots) and the shapes of these eigenvectors after multiplying with R_D (blue circles), for the 32×1 problem.

original iteration matrix R_F (given by red dots), the deflation technique shifts the two problematic eigenvalues to 1. However, two other eigenvalues are enlarged, as the piecewise constant deflation vectors may not properly approximate the steep gradients near the boundaries of the problematic eigenvectors.

Figure 10 displays by red dots the first four eigenvectors of iteration matrix R_D , which is based on the FFT + SD smoother. The blue circles in the figure show the shapes of these eigenvectors after multiplication by R_D . The two eigenvectors in (a) and (b) show symmetry and anti-symmetry. They are reduced but still contain steep gradients at the boundaries. The other two in (c) and (d) come from the piecewise constant deflation vectors, and they are preserved because the corresponding eigenvalues are 1.

Consider the computational work of the FFT + SD smoother. First, we look at the cost of an MVP with deflation matrix P , that is, $P\mathbf{x}$. Assume matrix Z consisting of m columns with $m \ll n$. Multiplying by Z or Z^T involves n operations.

- The inverse $E_{m \times m}^{-1}$ given in Equation (16) can be computed beforehand. In fact, in Equation (17), the MVP of A involves two MVPs by FFTs with complexity $\mathcal{O}(n \log(n))$. As $m \ll n$, the cost for $E_{m \times m}^{-1}$ can be ignored.
- For an MVP $\mathbf{y} := Q\mathbf{x}$ according to (16), $Z^T\mathbf{x}$ requires n operations. The cost for $E^{-1}(Z^T\mathbf{x})$ can be ignored, and $ZE^{-1}(Z^T\mathbf{x})$ requires n operations. Hence, one MVP with Q is $\mathcal{O}(n)$ complexity.
- For the MVP $P\mathbf{x}$ in (15), $\bar{A}Q\mathbf{x} = \bar{A}\mathbf{y}$ requires two MVPs, involving $\mathcal{O}(n \log(n))$ operations.

Therefore, $\mathcal{O}(n \log(n))$ operations are needed for one MVP $P\mathbf{x}$, which, roughly speaking, involves two MVPs in total. The FFT + SD smoother is defined by the scheme as

$$\mathbf{p}^{k+1} = (I - PMA)\mathbf{p}^k + PM\mathbf{u}, \quad (18)$$

where $(I - PMA)\mathbf{p}^k$ costs four MVPs, with two for matrix P and two for MA . The computation of $PM\mathbf{u}$ requires three MVPs; however, on the finest grid, it can be computed in advance.

4.2. Row sum modification

The second technique for improvement comes from the fact that the row sum of matrix A^{-1} is related to the elastic energy $J(\mathbf{p}) := \frac{1}{2}\mathbf{p}^T A\mathbf{p} - \mathbf{p}^T \mathbf{u}$, because the solution $\mathbf{p} = A^{-1}\mathbf{u}$ minimizes energy. It is expected that matrix M in the FFT smoother is an accurate approximation of A^{-1} . We compare the row sums of these two matrices in Figure 11 for the 32×1 problem.

We denote the row sums of any matrix A by $\text{sum}(A)$. It can be found from this figure that, first of all, $\text{sum}(A^{-1})$ exhibits almost constant values except for the rows at boundaries. Second, they are smaller than $\text{sum}(M)$ with especially large differences at the domain boundaries. It is due to the large difference in the boundary columns of A^{-1} and M , seen in Figure 3. This can be explained from a mechanics point of view, as follows.

We number the elements in contact area from the left to the right by 1 to n . When computing the coefficients \mathbf{m}^K for M by the FFT technique, the contact area is ‘extended’ into an infinite domain. Suppose that there is one ghost element numbered by -1 at the left, next to boundary element 1. According to the coefficient pattern in M , to achieve a unit deformation at element 1, a large positive traction on element 1 and negative tractions on other elements should be arranged. Element -1 also obtains a negative traction and contributes negative deformation to element 1. On the other hand, no ghost elements are present in A^{-1} . Hence, the elements at the boundaries do not receive negative contributions of deformation from them, and less traction values are required at these boundary elements to have unit deformation. This boundary effect gives rise to the difference between A^{-1} and M .

A modification of M is based on the aforementioned observation. The technique is to shift $\text{sum}(M)$ to a constant level, as $\text{sum}(A^{-1})$ appears constant at the interior rows. As $\text{sum}(M)$ is larger than $\text{sum}(A^{-1})$, the constant is chosen to be the minimum of $\text{sum}(M)$, which is the nearest to $\text{sum}(A^{-1})$. Shifting $\text{sum}(M)$ can be performed by only modifying diagonal entries, as they contribute most. Therefore, we will iterate with a shifted matrix \bar{M} , defined as

$$\bar{M} = M - \text{diag}(\text{sum}(M) - \gamma), \quad (19)$$

where $\text{diag}(\mathbf{x})$ is a diagonal matrix with \mathbf{x} the main diagonal entries, and the scalar

$$\gamma = \min[\text{sum}(M)]. \quad (20)$$

This leads to a matrix \bar{M} , with $\text{sum}(\bar{M}) = \gamma$.

To investigate the quality of the choice of γ in (20), we perform a numerical experiment and take 32×1 problem, and plot the spectral radii of the shifted iteration matrices $R_R = I - \bar{M}A$ with

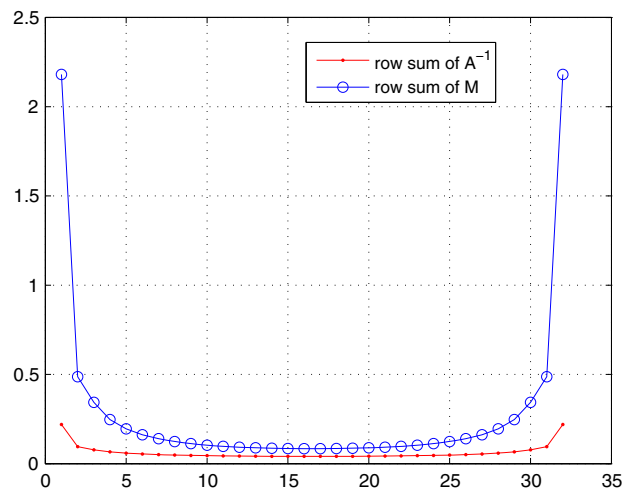


Figure 11. Row sums of A^{-1} (denoted by red dots) and of M (denoted by blue circles), for the 32×1 problem.

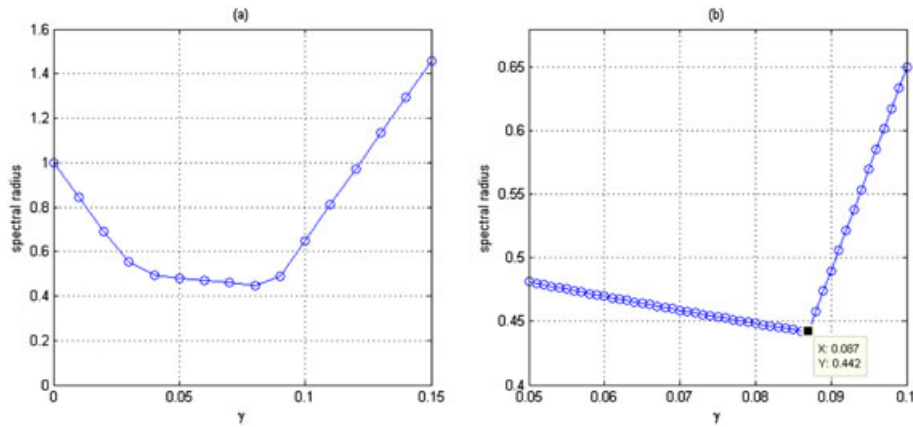


Figure 12. FFT + RSM smoother: the spectral radii of iteration matrices $R_R = (I - \bar{M}A)$ with (a) $\gamma \in [0, 0.15]$, for the 32×1 problem. (b) The situation around $\gamma = 0.08$ is displayed.

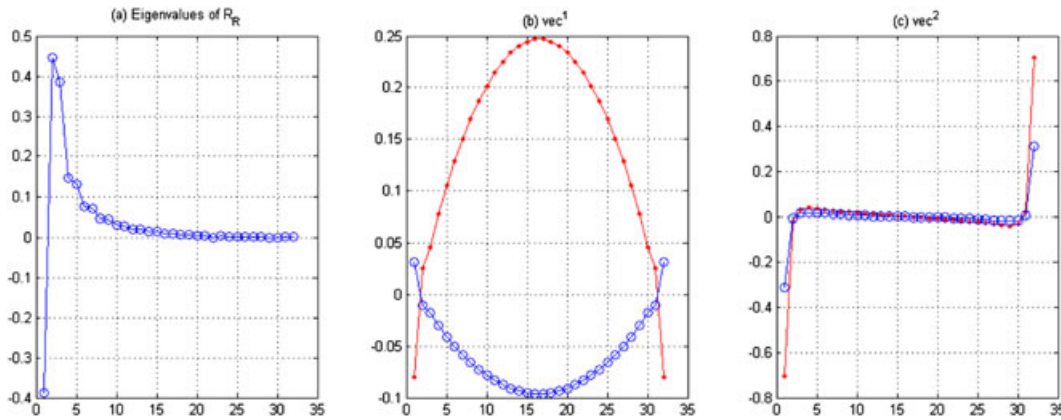


Figure 13. FFT + RSM smoother: (a) eigenvalues of the iteration matrix of R_R for the 32×1 problem. (b) The first eigenvector. (c) The second eigenvector.

the values $\gamma \in [0, 0.15]$, in Figure 12(a). As can be seen, $\gamma \approx 0.08$ results in the smallest spectral radius, and we zoom in around this value in Figure 12(b), where $\gamma = 0.087$ appears an optimal value. Based on Equation (20), we obtain $\gamma = 0.0837$, which is very close to this optimal value. Hence, we expect that this technique may work well.

Figure 13(a) presents the eigenvalues of iteration matrix $R_R = (I - \bar{M}A)$ with \bar{M} defined by Equation (19). All eigenvalues are now smaller than 1. The first eigenvector of R_R is denoted by red dots in Figure 13(b), where the blue circles represent its shape after multiplying by R_R . The function is smooth and can be dealt with by the coarse grid correction. Figure 13(c) shows the second eigenvector. It still exhibits a certain steepness at the boundaries.

Considering the computational cost, we check definition (19) again. We write the coefficients of M as $\mathbf{m} = [m_{-(n-1)}, \dots, m_{-1}, m_0, m_1, \dots, m_{(n-1)}]$. Here, $sum(M)$ can be computed in an efficient way because of symmetry of \mathbf{m} and the Toeplitz structure of M . First of all, we compute the first row sum f_1 , involving $n - 1$ operations. For the rows $i = 2, \dots, n$, the sum f_i can be computed recursively by $f_i = f_{i-1} - m_{n-(i-1)} + m_{i-1}$, where two operations are required for each f_i . So, $3(n - 1)$ operations are needed to determine $sum(M)$. The arithmetic complexity is reduced from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$. An alternative that has $\mathcal{O}(n \log(n))$ complexity is to multiply M by a vector with all entries equal to 1, using the FFT technique.

In each MVP with \bar{M} , the first part M needs $\mathcal{O}(n \log(n))$ operations by FFTs because of the Toeplitz structure. The second part requires $\mathcal{O}(n)$ operations because it is a diagonal matrix. Hence,

$\mathcal{O}(n \log(n))$ operations are required for this MVP. The iteration scheme of the FFT + RSM smoother reads as follows:

$$\mathbf{p}^{k+1} = (I - \bar{M}A) \mathbf{p}^k + \bar{M}\mathbf{u}. \quad (21)$$

The first term at the right-hand side involves two MVPs and the second involves one MVP, which can be computed in advance on the target grid. So in one FFT + RSM smoothing step, two MVPs are required on the target grid and three MVPs on the coarser grids. The complexity $\mathcal{O}(n \log(n))$ is due to the Toeplitz structure.

5. NUMERICAL RESULTS

In this section, we solve three test problems, including test 1 presented in Section 2.5. Test 2 is concerned with a different material as well as a different right-hand side of $A\mathbf{p} = \mathbf{u} = -\delta t \mathbf{w}$. It resembles a 3D case, with spatially varying rigid slip ($\delta t \mathbf{w}$). This is due to so-called spin creepage, which results from relative rotation of the cylinder around the normal to the contact area [44]. Solving test 2 serves as a preparation for future work on 3D problems.

Test 3 has the same setting as test 1, except that the plane is not a smooth but a wavy surface. It resembles a rough surface contact problem, which has irregular domains as contact areas. This type of problems stems from railway applications [45, 46]. In test 3, the true contact area consists of two strips, which destroys the Toeplitz structure of the resulting system, which is one of the reasons why we do not consider existing direct Toeplitz solvers for our contact problems.

Summarizing:

- Test 1: the rigid slip ($\delta t \mathbf{w}$) is constant.
- Test 2: the contacting bodies are rubber with shear modulus and Poisson ratio being $G = 0.3 \text{ N/mm}^2$ and $\mu = 0.49$, respectively. The rigid slip has a linear relation with position: ($\delta t \mathbf{w}$) = $-\mathbf{x}$.
- Test 3: the contact area is $[-50, 50]$ mm in the y -direction but consists of two intervals in the x -direction, that is, $[-4, -2] \cup [0, 4]$ mm, due to the wavy surface.

We first solve tests 1 and 2. As test 3 loses the Toeplitz structure, it is discussed separately at the end of this section.

Our solver is the MG method with the FFT smoother, enhanced by either SD or RSM. The initial guess is chosen randomly. The iterations terminate when $\frac{\|\mathbf{u} - A\mathbf{p}^k\|_{rms}}{\|\mathbf{u}\|_{rms}} \leq 10^{-8}$, where we define the norm as $\|\mathbf{x}\|_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}$. The coarsest grid is chosen to be 2×1 . To investigate the performance, we use the convergence factor as a measure. It is defined as follows [39]: $\hat{q}^k = \sqrt[k]{\frac{\|\mathbf{r}^k\|_{rms}}{\|\mathbf{r}^0\|_{rms}}}$, where \mathbf{r}^k is the residual after the k th iteration.

5.1. Results by MG with the FFT + SD smoother

The numbers of iterations and the convergence factors by MG with the FFT + SD smoother for the first two tests are shown in Tables II and III, respectively. A $V(1,1)$ -cycle does not converge with the components chosen, but V -cycles with one smoothing step perform well. Usually, a $V(1,1)$ -cycle converges faster than a V -cycle with one smoothing step, which is not the case here. One reason is the inefficient approximation of some eigenvectors of \bar{A} by piecewise constant deflation vectors, which are not removed from the residual and influence the performance of the smoother.

Table II. MG with the FFT + SD smoother for test 1: the numbers of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
$V(1,0)$	6 (0.0363)	6 (0.0355)	5 (0.0181)	6 (0.0367)	7 (0.0641)	7 (0.0626)
$V(0,1)$	6 (0.0348)	5 (0.0226)	5 (0.0230)	5 (0.0233)	6 (0.0397)	7 (0.0678)

Table III. MG with the FFT + SD smoother for test 2: the numbers of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
$V(1,0)$	11 (0.1452)	11 (0.1492)	12 (0.1619)	12 (0.1765)	13 (0.1957)	14 (0.2031)
$V(0,1)$	10 (0.1092)	10 (0.1098)	10 (0.1094)	10 (0.1093)	10 (0.1095)	10 (0.1140)

5.2. Results by MG with the FFT + RSM smoother

The results by MG with the FFT + RSM smoother for the first two tests are presented in Tables IV and V, respectively. As can be seen, all three V -cycles converge fast and show mesh independence. Moreover, the $V(1,1)$ -cycle requires almost half of the iterations that are needed by the other two V -cycles. Also for other right-hand side functions, very similar results were obtained. The FFT + RSM smoother works efficiently in the basic MG framework defined in Section 3.3.

Regarding the computational cost, we consider the number of work units, which are defined as *the work of one MVP on the finest grid*. (All MVPs are performed by $\mathcal{O}(n \log(n))$ operations using FFTs due to the Toeplitz structure.) This work unit is halved on a coarser grid, due to the standard coarsening in Figure 8. In one $V(1,0)$ - or $V(0,1)$ -cycle on the finest grid, three MVPs are required, including two for FFT + RSM smoothing and one for defect computation. Similarly, five MVPs are involved in one $V(1,1)$ -cycle. One more MVP is required on coarser grid levels. The total number of work units is obtained by multiplying the cost in one cycle by the number of cycles.

We take the $2^{20} \times 1$ problem as an example, which consists of more than one million unknowns. Table VI shows the (rounded) number of work units by MG with FFT + RSM smoother for tests 1 and 2. The $V(0,1)$ -cycle appears favorable. Codes are written in Matlab (MathWorks, Natick, Massachusetts) 8.0 (R2012b) on Windows 7 (Microsoft Corporation, Redmond, Washington) 64-bit platform with Intel(R) (Intel Corporation, Santa Clara, California) Core(TM)2 Duo CPU, E8500@3.16 GHz, 3.17 GHz. The CPU time by $V(0,1)$ -cycle for these two tests is around 20 s.

Table IV. MG with the FFT + RSM smoother for test 1: the numbers of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
$V(1,1)$	4 (0.0053)	3 (0.0019)	3 (0.0015)	3 (0.0012)	3 (0.0010)	3 (0.0008)
$V(1,0)$	6 (0.0430)	6 (0.0388)	5 (0.0244)	5 (0.0215)	5 (0.0190)	4 (0.0089)
$V(0,1)$	6 (0.0370)	5 (0.0231)	5 (0.0178)	5 (0.0083)	4 (0.0070)	4 (0.0070)

Table V. MG with the FFT + RSM smoother for test 2: the numbers of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
$V(1,1)$	4 (0.0051)	4 (0.0044)	4 (0.0037)	4 (0.0031)	3 (0.0010)	3 (0.0008)
$V(1,0)$	8 (0.0630)	7 (0.0486)	7 (0.0444)	6 (0.0308)	6 (0.0278)	6 (0.0251)
$V(0,1)$	7 (0.0468)	7 (0.0427)	6 (0.0294)	6 (0.0264)	5 (0.0153)	5 (0.0135)

Table VI. MG with the FFT + RSM smoother: the (rounded) number of work units for a $2^{20} \times 1$ problem in tests 1 and 2.

Multigrid cycle	Test 1	Test 2
$V(1,1)$	37	37
$V(1,0)$	29	43
$V(0,1)$	29	36

5.3. Results by the FFT + RSM used as a solver

The iteration matrix R_R had an eigenvector that was not smoothly varying near the boundary, as shown in Figure 13(c). We thus expect that standard coarse grid correction does not help significantly for convergence acceleration. Here, we also apply the FFT + RSM technique as an iterative solver *on only one grid*. The results are given in Table VII for the two tests, showing kind of stable iteration numbers as the problem size becomes larger.

Regarding the computational cost, we again take the $2^{20} \times 1$ problem as an example. As two MVPs are involved in each iteration, the numbers of work units are 58 and 66 for tests 1 and 2, respectively. Compared to the numbers in Table VI using MG with the FFT + RSM smoother, we see that the MG is favorable.

5.4. Test 3

The contact area in test 3 consists of two strips, $[-4, -2]$ and $[0, 4]$ in the x -direction and $[-50, 50]$ in the y -direction. We define the *potential contact area* as $[-4, 4] \cup [-50, 50]$, which contains these two strips. The influence coefficient matrix A on this potential contact area is still Toeplitz. The true coefficient matrix A_r for these two strips is obtained by removing from A the rows and columns corresponding to the elements that are not in these two strips. Hence, A_r is not a Toeplitz matrix anymore. However, it is still a dense and SPD matrix.

To benefit from the FFT technique, we still employ a Toeplitz matrix A when we implement MVPs $\mathbf{u}_r = A_r \mathbf{x}$. This is performed as explained in [29]. First, we put the entries of \mathbf{x} that are not in contact to zero and apply the FFT to compute $A\mathbf{x} = \mathbf{u}$. Then we set the entries of \mathbf{u} that are not in the two contact strips to zero, which leads to the desired result \mathbf{u}_r .

As we lost the Toeplitz structure, the RSM approach requires the following adaptation:

$$\bar{M}_{imp} = M_r - \text{diag}(\text{sum}(M_r) - \gamma), \tag{22}$$

where $\gamma = \min[\text{sum}(M)]$ is again the minimal row sum of the original preconditioner M , which is constructed by the FFT approach on the potential contact area. The matrix M_r is obtained from M in the same way as we obtain A_r from A . The resulting method is still $\mathcal{O}(n \log(n))$.

Table VIII presents the results by MG with the FFT + RSM smoother. The three V -cycles converge rapidly and show mesh-independent MG convergence. Table IX gives the results by the FFT + RSM solver, which also indicates highly satisfactory convergence and stable iteration numbers on different grids.

Regarding the computational costs, we again take $2^{20} \times 1$ as the example. The number of work units for $V(1,1)$ -, $V(1,0)$ -, and $V(0,1)$ -cycles are 37, 29, and 29, respectively. The CPU time for the $V(0,1)$ -cycle is around 21 s. The FFT + RSM solver requires 78 work units and 58 s CPU time. MG with the FFT + RSM smoother compares favorably.

Table VII. The FFT + RSM solver: the number of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
Test 1	28 (0.5085)	28 (0.5109)	28 (0.5126)	28 (0.5139)	28 (0.5149)	29 (0.5194)
Test 2	33 (0.5342)	34 (0.5370)	34 (0.5379)	34 (0.5381)	33 (0.5346)	33 (0.5346)

Table VIII. MG with the FFT + RSM smoother for test 3: the numbers of iterations with the convergence factors between brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
$V(1,1)$	4 (0.0057)	4 (0.0049)	3 (0.0017)	3 (0.0014)	3 (0.0011)	3 (0.0009)
$V(1,0)$	6 (0.0454)	6 (0.0409)	6 (0.0369)	5 (0.0230)	5 (0.0202)	4 (0.0096)
$V(0,1)$	6 (0.0392)	6 (0.0352)	5 (0.0217)	5 (0.0190)	4 (0.0090)	4 (0.0076)

Table IX. FFT + RSM solver for test 3: the iteration numbers with convergence factors in brackets for different discretizations.

Discretization	$2^{15} \times 1$	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
#it(conv. fact.)	36 (0.5971)	36 (0.5975)	36 (0.5977)	36 (0.5980)	37 (0.6004)	37 (0.6021)

6. CONCLUSION

In this paper, we investigated an FFT smoother in an MG method, to solve a simplified 2D frictional contact problem. The discretization of the integral equation gives rise to a linear system with a coefficient matrix being dense, SPD, and of Toeplitz structure. A Toeplitz preconditioner M is used in the smoother, whose construction and implementation are relatively cheap by the use of the FFT technique. This smoother reduces many components of the error but enlarges several smooth components. In order to overcome this drawback, two techniques have been studied: SD and RSM. MG with the FFT + RSM smoother shows a rapid convergence and mesh independence in the numerical tests, particularly in test 3 where the Toeplitz structure is lost. In addition, the FFT + RSM smoother is applied as a stand-alone solver, which also converges well. Moreover, both solvers are $\mathcal{O}(n \log(n))$ methods.

We are currently working on the extension of the FFT method for 3D problems in which a BTTB matrix should be solved.

ACKNOWLEDGEMENTS

Jing Zhao is grateful for the financial support by the China Scholarship Council (CSC).

REFERENCES

1. Johnson KL. *Contact Mechanics*. Cambridge University Press: UK, 1985.
2. Kalker JJ. *Three-Dimensional Elastic Bodies in Rolling Contact*. Kluwer Academic Publisher: The Netherlands, 1990.
3. Enblom R. Deterioration mechanisms in the wheel–rail interface with focus on wear prediction: a literature review. *Vehicle System Dynamics* 2009; **47**(6):661–700.
4. Rabinowicz E. *Friction and Wear of Materials*, 2nd edition. Wiley: New York, 1995.
5. Grassie SL. Rolling contact fatigue on the British railway system: treatment. *Wear* 2005; **258**(7):1310–1318.
6. Van Beek A. *Advanced Engineering Design: Lifetime Performance and Reliability*. Delft University of Technology: The Netherlands, 2006.
7. Popov VL. *Contact Mechanics and Friction: Physical Principles and Applications*. Springer: Berlin, 2010.
8. Kikuchi N, Oden JT. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 1987.
9. Wriggers P. *Computational Contact Mechanics*. Springer-Verlag: Berlin, Heidelberg, 2006.
10. Krause R. A nonsmooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency. *SIAM Journal on Scientific Computing* 2009; **31**(2):1399–1423.
11. Andersson T. The boundary element method applied to two-dimensional contact problems with friction. *Proceedings of the 3rd international seminar on recent advances in boundary element methods*, Springer, Berlin, 1981.
12. Man KW. *Contact Mechanics Using Boundary Elements*. Computational Mechanics Publisher: Boston MA, 1994.
13. Vollebregt EAH. User’s guide for CONTACT, Vollebregt & Kalker’s rolling and sliding contact model. *Technical Report TR09-03*, version 13.1, VORtech, 2013.
14. Vollebregt EAH. A Gauss–Seidel type solver for special convex programs, with application to frictional contact mechanics. *Journal of Optimization Theory and Applications* 1995; **87**(1):47–67.
15. Hansen PC, Nagy JG, O’leary DP. *Deblurring Images: Matrices, Spectra, and Filtering*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 2007.
16. Scharf LL. *Statistical Signal Processing. Detection, Estimation, and Time Series Analysis*. Addison-Wesley: Reading MA, 1991.
17. Ammar GS. Classical foundations of algorithms for solving positive definite Toeplitz equations. *Calcolo* 1996; **33**(1-2):99–113.
18. Strang G. A proposal for Toeplitz matrix calculations. *Studies in Applied Mathematics* 1986; **74**(2):171–176.
19. Huckle T. Circulant and skew circulant matrices for solving Toeplitz matrix problems. *SIAM Journal on Matrix Analysis and Applications* 1992; **13**(3):767–777.

20. Chan RHF, Jin XQ. *An Introduction to Iterative Toeplitz Solvers*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 2007.
21. Gray RM. *Toeplitz and Circulant Matrices: A Review*. Now Publisher Inc.: Hanover, USA, 2006.
22. Chan RH, Chang QS, Sun HW. Multigrid method for ill-conditioned symmetric Toeplitz systems. *SIAM Journal on Scientific Computing* 1998; **19**(2):516–529.
23. Donatelli M. An algebraic generalization of local Fourier analysis for grid transfer operators in multigrid based on Toeplitz matrices. *Numerical Linear Algebra with Applications* 2010; **17**(2-3):179–197.
24. Bolten M, Donatelli M, Huckle T. Aggregation-based multigrid methods for circulant and Toeplitz matrices, Bergische Universität Wuppertal, 2012. Preprint BUW-IMACM 12/10.
25. Capizzano SS. Convergence analysis of two-grid methods for elliptic Toeplitz and PDEs matrix-sequences. *Numerische Mathematik* 2002; **92**(3):433–465.
26. Venner CH, Lubrecht AA. *Multi-level Methods in Lubrication*. Elsevier Science: Amsterdam, The Netherlands, 2000.
27. Stanley HM, Kato T. An FFT-based method for rough surface contact. *Journal of Tribology* 1997; **119**(3):481–485.
28. Zhao J, Vollebregt EAH, Oosterlee CW. A full multigrid method for linear complementarity problems arising from elastic normal contact problems, 2012. Student competition paper for the 12th Copper Mountain conference, USA.
29. Vollebregt EAH. A new solver for the elastic normal contact problem using conjugate gradients, deflation and an FFT-based preconditioner. *Journal of Computational Physics* 2014; **257**:333–351.
30. Vollebregt EAH, Schuttelaars HM. Quasi-static analysis of two-dimensional rolling contact with slip-velocity dependent friction. *Journal of Sound and Vibration* 2012; **331**(9):2141–2155.
31. Hertz H. Über die berührung fester elastischer Körper. *Journal für Reine und Angewandte Mathematik* 1882; **92**:156–171.
32. Von Petersdorff T, Stephan EP. Multigrid solvers and preconditioners for first kind integral equations. *Numerical Methods for Partial Differential Equations* 1992; **8**(5):443–450.
33. Brandt A, Lubrecht AA. Multilevel matrix multiplication and fast solution of integral equations. *Journal of Computational Physics* 1990; **90**(2):348–370.
34. Hackbusch W. *Multi-grid Methods and Applications*. Springer-Verlag: Berlin, Heidelberg, 1985.
35. Atkinson KE. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press: UK, 1997.
36. Hemker PW, Schippers H. Multiple grid methods for the solution of Fredholm integral equations of the second kind. *Mathematics of Computation* 1981; **36**(153):215–232.
37. Van Loan C. *Computational Frameworks for the Fast Fourier Transform*. Society for Industrial and Applied Mathematics: Philadelphia, USA, 1987.
38. Ng MK. *Iterative Methods for Toeplitz Systems*. Oxford University Press: UK, 2004.
39. Trottenberg U, Oosterlee CW, Schüller A. *Multigrid*. Academic Press: London, 2000.
40. Tang KT. *Mathematical Methods for Engineers and Scientists 3: Fourier Analysis, Partial Differential Equations and Variational Methods*, Vol. 3. Springer-Verlag: Berlin, Heidelberg, 2007.
41. Nicolaidis RA. Deflation of conjugate gradients with applications to boundary value problems. *SIAM Journal on Numerical Analysis* 1987; **24**(2):355–365.
42. Frank J, Vuik C. On the construction of deflation-based preconditioners. *SIAM Journal on Scientific Computing* 2001; **23**(2):442–462.
43. Tang JM. Two-level preconditioned conjugate gradient methods with applications to bubbly flow problems. *Ph.D. Thesis*, Delft University of Technology, Delft, The Netherlands, 2008.
44. Ayasse J-B, Chollet H. Wheel–rail contact. In *Handbook of Railway Vehicle Dynamics*, Iwnicki S (ed.). chap. 4, CRC Press: Boca Raton, 2006; 85–120.
45. Bucher F, Knothe K, Theiler A. Normal and tangential contact problem of surfaces with measured roughness. *Wear* 2002; **253**(1):204–218.
46. Kalker JJ, Dekking FM, Vollebregt EAH. Simulation of rough, elastic contacts. *Transaction ASME Journal of Applied Mechanics* 1997; **64**(2):361–368.