Found Comput Math (2018) 18:45–95
https://doi.org/10.1007/s10208-016-9332-x

FOUNDATIONS OF COMPUTATIONAL MATHEMATICS
The Journal of the Society for the Foundations of Computational Mathematics

Fast Structured Matrix Computations: Tensor Rank and Cohn–Umans Method

Ke Ye[1] · **Lek-Heng Lim**[2]

Received: 1 January 2016 / Revised: 16 June 2016 / Accepted: 22 August 2016 /
Published online: 26 September 2016

Abstract We discuss a generalization of the Cohn–Umans method, a potent technique developed for studying the bilinear complexity of matrix multiplication by embedding matrices into an appropriate group algebra. We investigate how the Cohn–Umans method may be used for bilinear operations other than matrix multiplication, with algebras other than group algebras, and we relate it to Strassen's tensor rank approach, the traditional framework for investigating bilinear complexity. To demonstrate the utility of the generalized method, we apply it to find the fastest algorithms for forming structured matrix–vector product, the basic operation underlying iterative algorithms for structured matrices. The structures we study include Toeplitz, Hankel, circulant, symmetric, skew-symmetric, f-circulant, block Toeplitz–Toeplitz block, triangular Toeplitz matrices, Toeplitz-plus-Hankel, sparse/banded/triangular. Except for the case of skew-symmetric matrices, for which we have only upper bounds, the algorithms derived using the generalized Cohn–Umans method in all other instances are the fastest possible in the sense of having minimum bilinear complexity. We also apply this framework to a few other bilinear operations including matrix–matrix, commutator, simultaneous matrix products, and briefly discuss the relation between tensor nuclear norm and numerical stability.

Communicated by Nicholas Higham.

✉ Lek-Heng Lim
lekheng@galton.uchicago.edu

Ke Ye
kye@galton.uchicago.edu

[1] Department of Statistics, University of Chicago, Chicago, IL 60637, USA

[2] Computational and Applied Mathematics Initiative, Department of Statistics, University of Chicago, Chicago, IL 60637, USA

Keywords Bilinear complexity · Tensor rank · Tensor nuclear norm · Cohn–Umans method · Structured matrix–vector product · Stability · Sparse and structured matrices

Mathematics Subject Classification 15B05 · 65F50 · 65Y20 · 13P25 · 22D20

1 Introduction

In this article, we systematically study the design of fast, possibly fastest, algorithms for a variety of operations involving structured matrices, as measured by the *bilinear complexity* of the problem. Roughly speaking, the bilinear complexity of an algorithm for a problem that can be cast as the evaluation of a bilinear map is the number of multiplications required in the algorithm; the bilinear complexity of the problem is then that of an algorithm with the lowest bilinear complexity [6, Chapter 14]. This notion of complexity is best known for its use in quantifying the speed of matrix–matrix product and matrix inversion in the work of Strassen [40], Coppersmith–Winograd [13], Vassilevska Williams [46], and many others. The current record, due to Le Gall [30], for the asymptotic bilinear complexity of $n \times n$ matrix–matrix product for unstructured matrices is $O(n^{2.3728639})$. Roughly speaking, the asymptotic bilinear complexity of a problem dependent on n refers to its bilinear complexity when n is sufficiently large.

The algorithms that we study in this article will be for the following operations: (1) matrix–vector product, (2) matrix–matrix product, and (3) commutator product:

$$(A, x) \mapsto Ax, \qquad (A, B) \mapsto AB, \qquad (A, B) \mapsto AB - BA,$$

where A and B are structured matrices and x is a vector, of appropriate dimensions so that the products are defined.

The structured matrices studied in this article include: (1) sparse (including banded and triangular), (2) symmetric, (3) skew-symmetric, (4) Toeplitz, (5) Hankel, (6) circulant, (7) f-circulant and skew-circulant, (8) block Toeplitz–Toeplitz block (BTTB) and more generally any block structured matrices with structured blocks, (9) triangular Toeplitz and its analogues for Hankel and circulant matrices, (10) sum of Toeplitz and Hankel. We provide algorithms of optimal bilinear complexity for all except the skew-symmetric case (for which we only have upper bounds). The optimal bilinear complexity for the Toeplitz and triangular Toeplitz matrix–vector product are well-known, due to Bini and Capovani [2], but we will obtain them using a different method (generalized Cohn–Umans) that applies more generally to all classes of structured matrices discussed here.

We will examine two different approaches: the Strassen tensor rank approach [42,43], and the Cohn–Umans group theoretic approach [8–10], as well as the relations between them. Our study gives a generalization of the Cohn–Umans approach in two regards: a generalization from matrix–matrix product to arbitrary bilinear operations, and a generalization from (a) group algebras (e.g., Sect. 17) to arbitrary algebras including (b) cohomology rings of manifolds (e.g., Sect. 11), (c) coordinate rings of schemes (e.g., Sect. 11) and varieties (e.g., Sect. 13), (d) polynomial identity rings (e.g., Sect. 16). We will provide the equivalent of their 'triple product property' in

these more general contexts. The idea of considering algebras other than group algebras was already in [10], where the authors proposed to use adjacency algebras of coherent configurations. These may be viewed as a generalization of group algebras and are in particular semisimple, i.e., isomorphic to an algebra of block diagonal matrices. Our generalization goes further in that the algebras we use may contain nilpotents and thus cannot be semisimple (e.g., Sect. 11); in fact they may not be associative algebras (e.g., Sect. 16), may not be algebras (e.g., Sect. 15), and may not even be vector spaces (e.g., Example 6).

We hope to convince our readers, by way of a series of constructions involving various structured matrices and various bilinear operations, that this generalization of Cohn–Umans method could allow one to systematically uncover fast algorithms, and these could in turn be shown to be the fastest possible (in terms of bilinear complexity) via arguments based on the Strassen tensor rank approach. For instance, we will see in Sect. 14 that the fastest possible algorithm for multiplying a symmetric matrix to a vector involves first writing the symmetric matrix as a sum of Hankel matrices of decreasing dimensions bordered by zeros. For example, a 4×4 symmetric matrix would have to be decomposed into

$$\begin{bmatrix} a & b & c & d \\ b & e & f & g \\ c & f & h & i \\ d & g & i & j \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ b & c & d & g \\ c & d & g & i \\ d & g & i & j \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & e-c & f-d & 0 \\ 0 & f-d & e-c & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & h-g-e+c & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

This is highly nonobvious to us. We would not have been able to find this algorithm without employing the generalized Cohn–Umans approach.

The main focus of our article will be the *matrix–vector product* for various structured matrices since these form the fundamental building blocks of most modern iterative algorithms for problems involving structured matrices: linear systems [7,34], least-squares problems [3,34], eigenvalue problems [47], evaluating analytic functions with matrix arguments [21], etc. On the other hand, problems requiring matrix–matrix product of structured matrices are relatively uncommon; one reason being that the most common structured matrices (symmetric, Toeplitz, Hankel, etc; in fact all but circulant) are not closed under matrix–matrix products. Explicit pseudocodes for all structured matrix–vector product algorithms appearing in this article may be found in [49].

1.1 Why Minimize Multiplications?

In modern computer processors, there is no noticeable difference in the latency of addition and multiplication [23, Tables 14-1 and 15-6]. So the reader might wonder why bilinear complexity continues to be of relevance. We provide three reasons below.

The first reason is that such algorithms apply when we have matrices in place of scalars. We illustrate this with a simple example, Gauss's method for multiplying two complex numbers [26, Section 4.6.4]. Let $a, b, c, d \in \mathbb{R}$. Then the usual method

$$(a + ib)(c + id) = (ac - bd) + i(ad + bc)$$

requires four real multiplications and two real additions but Gauss's method

$$(a + ib)(c + id) = (ac - bd) + i[(a + b)(c + d) - ac - bd] \tag{1}$$

requires three real multiplications and five real additions. If the costs of addition and multiplication are roughly the same, then Gauss's method is a poor way for multiplying complex numbers. However, the usefulness of Gauss's method comes into view when we multiply complex *matrices* [20, Chapter 23], i.e., when we do

$$(A + iB)(C + iD) = (AC - BD) + i[(A + B)(C + D) - AC - BD]$$

where $A, B, C, D \in \mathbb{R}^{n\times n}$. Now Gauss's method requires three *matrix multiplications* instead of four. Addition and multiplication of scalars may well have similar computational costs, but multiplication of $n \times n$ matrices is by any measure vastly more expensive[1] than addition of $n \times n$ matrices. This observation applies more generally. For example, Strassen's algorithm for the product of 2×2 matrices [40] only becomes practically useful when it is applied (recursively) to the product of 2×2 *block matrices* [20, Chapter 23].

A second reason is that the preceding comparison of addition and multiplication implicitly assumes that we are using the traditional measure of computational cost, i.e., time complexity, but other measures, e.g., energy consumption, number of gates, code space, etc, have become increasingly important. For instance, a multiplier requires many more gates than an adder (e.g., 2200 gates for an 18-bit multiplier versus 125 gates for an 18-bit adder [27]), which translates into more wires and transistors on a microchip and also consumes more energy.

A third reason is that while the latencies of addition and multiplication are comparable on a general purpose CPU, it is important to remember that arithmetic is performed on other microchips as well, e.g., ASIC, DSP, FPGA, GPU, motion coprocessor, etc, where the latency of multiplication may be substantially higher than that of addition. Moreover, our second reason also applies in this context.

1.2 Overview

We begin by introducing the central object of this article, the structure tensor of a bilinear operation, and discuss several examples in Sect. 2. This is followed by a discussion of tensor rank and the closely related notion of border rank in Sect. 3, allowing us to define bilinear complexity rigorously as the rank of a structure tensor. We prove several results regarding tensor rank and border rank that will be useful later when we need to determine these for a given structure tensor. We end the section with a brief discussion of numerical stability and its relation to the nuclear norm of the structure tensor.

[1] Even if the exponent of matrix multiplication turns out to be 2; note that this is asymptotic.

In Sect. 4, we examine the structure tensor in the special case where the bilinear operation is the product operation in an algebra and prove a relation between tensor ranks of the respective structure tensors when one algebra is mapped into another. This provides partial motivation for the generalized Cohn–Umans method in Sect. 5, where we first present the usual Cohn–Umans method as a commutative diagram of algebras and *vector space homomorphisms* (as opposed to homomorphisms of algebras), followed by a demonstration that the 'triple product property' is equivalent to the commutativity of the diagram. Once presented in this manner, the Cohn–Umans method essentially generalizes itself. As a first example, we show that the fast integer multiplication algorithms of Karatsuba et al. may be viewed as an application of the generalized Cohn–Umans method.

In the remainder of the article, we apply the generalized Cohn–Umans method to analyze a variety of structured matrix–vector products:

- sparse, banded, triangular: Sect. 6,
- circulant: Sect. 7,
- f-circulant, skew-circulant: Sect. 8,
- Toeplitz: Sect. 9,
- Hankel: Sect. 10,
- triangular Toeplitz/Hankel: Sect. 11,
- Toeplitz-plus-Hankel: Sect. 12,
- block Toeplitz–Toeplitz block and other multilevel structures: Sect. 13,
- symmetric: Sect. 14,
- skew-symmetric: Sect. 15.

Except for the case of skew-symmetric matrices, we obtain algorithms with optimum bilinear complexities for all structured matrix–vector products listed above. In particular we obtain the rank and border rank of the structure tensors in all cases but the last.

A reader who follows the developments in Sects. 7–15 will observe a certain degree of interdependence between these algorithms. For example, as we have mentioned earlier, the algorithm for symmetric matrix–vector product depends on that for Hankel matrix–vector product, but the latter depends on that for Toeplitz matrix–vector product, which in turn depends on that for circulant matrix–vector product. As another example of a somewhat surprising interdependence, in Sect. 16, we discuss an algorithm for the commutator product, i.e., $[A, B] = AB - BA$, for 2×2 matrices A, B based on the algorithm for 3×3 skew-symmetric matrix–vector product in Sect. 15. Yet a third example is that our algorithm for skew-circulant matrix–vector product in Sect. 8 turns out to contain Gauss's multiplication of complex numbers as a special case: (1) may be viewed as the product of a skew-circulant matrix in $\mathbb{R}^{2\times 2}$ with a vector in $\mathbb{R}^2$.

To round out this article, we introduce a new class of problems in Sect. 17 that we call 'simultaneous product' of matrices. The most natural problem in this class would be the simultaneous computation of AB and AB^{T} for a square matrix B, but we are unable to obtain any significant findings in this case. Nevertheless, we provide an impetus by showing that the closely related variants of simultaneously computing the pair of matrix products

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} g & h \\ e & f \end{bmatrix},$$

or the pair of matrix products

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} h & g \\ e & f \end{bmatrix},$$

can be obtained with just *eight* multiplications and that the resulting algorithms have optimum bilinear complexity. Note that computing the pair of products separately via Strassen's algorithm, which is optimum for 2×2 matrix–matrix product, would require 14 multiplications.

Throughout this article, we work over $\mathbb{C}$ for simplicity but our results hold for more general fields—quadratic, cyclotomic, infinite, or algebraically closed extensions of an arbitrary field (say, a finite field), depending on the context.

Results in Sects. 2–6 and 17 are independent of our choice of field with a few exceptions: (1) any discussion of Gauss's method is of course peculiar to $\mathbb{C}$ but generalizes to any quadratic extension of an arbitrary field; (2) the discussion of numerical stability in Sect. 3.2 require that we work over a subfield of $\mathbb{C}$ since they involve norms; (3) Winograd's theorem (Theorem 3) requires an infinite field; (4) Corollary 5 requires an algebraically closed field. The results in Sects. 7–14 for $n \times n$ structured matrices require that the field contains all nth roots of some element, usually 1 but sometimes -1 (for skew-circulant or skew-symmetric) or f (for f-circulant). Results in Sects. 15 and 16 require an algebraically closed field.

2 The Structure Tensor of a Bilinear Operation

A *bilinear operation* is simply a bilinear map $\beta : U \times V \to W$ where U, V, W are vector spaces over the same field, henceforth assumed to be $\mathbb{C}$. For example, the operation of forming a matrix–vector product is a bilinear operation $\beta : \mathbb{C}^{m\times n} \times \mathbb{C}^n \to \mathbb{C}^m$, $(A, x) \mapsto Ax$, since

$$\beta(aA + bB, x) = a\beta(A, x) + b\beta(B, x), \qquad \beta(A, ax + by) = a\beta(A, x) + b\beta(A, y).$$

Likewise for the operations of matrix–matrix product and commutator product.

A simple but central observation in the study of bilinear complexity is that every bilinear operation is characterized by a 3-tensor and that its tensor rank quantifies the complexity, as measured solely in terms of the number of multiplications, of the bilinear operation. We start by defining this 3-tensor.

Definition/Proposition 1 *Let $\beta : U \times V \to W$ be a bilinear map. Then there exists a unique tensor $\mu_\beta \in U^* \otimes V^* \otimes W$ such that given any $(u, v) \in U \times V$ we have*

$$\beta(u, v) = \mu_\beta(u, v, \cdot) \in W.$$

We call μ_β the structure tensor of the bilinear map β.

By the definition of tensor product, there is a one-to-one correspondence between the set of bilinear maps from $U \times V$ to W and the set of linear maps from $U \otimes V$ to W. Therefore we do not distinguish between a bilinear map $\beta : U \times V \to W$ and its corresponding linear map $\beta : U \otimes V \to W$ (and denote both by β).

In the special case when $U = V = W = \mathscr{A}$ is an algebra and the bilinear map $\beta : \mathscr{A} \times \mathscr{A} \to \mathscr{A}$, $(u, v) \mapsto uv$, is multiplication in $\mathscr{A}$. The structure tensor of β is called the *structure tensor of the algebra* $\mathscr{A}$, and is denoted by $\mu_{\mathscr{A}}$.

Example 1 (*Lie algebras*) Let $\mathfrak{g}$ be a complex Lie algebra of dimension n and let $\{e_1, \ldots, e_n\}$ be a basis of $\mathfrak{g}$. Let $\{e_1^*, \ldots, e_n^*\}$ be the corresponding dual basis defined in the usual way as

$$e_i^*(e_j) = \begin{cases} 1 & i = j, \\ 0 & i \neq j, \end{cases}$$

for all $i, j = 1, \ldots, n$. Then for each pair $i, j \in \{1, \ldots, n\}$,

$$[e_i, e_j] = \sum_{k=1}^{n} c_{ij}^k e_k,$$

for some constant numbers $c_{ij}^k \in \mathbb{C}$. The structure tensor of the Lie algebra $\mathfrak{g}$ is

$$\mu_{\mathfrak{g}} = \sum_{i,j,k=1}^{n} c_{ij}^k e_i^* \otimes e_j^* \otimes e_k \in \mathfrak{g}^* \otimes \mathfrak{g}^* \otimes \mathfrak{g}.$$

The constants c_{ij}^k are often called the *structure constants* of the Lie algebra and the hypermatrix [31]

$$[c_{ij}^k] \in \mathbb{C}^{n \times n \times n}$$

is the coordinate representation of $\mu_{\mathscr{A}}$ with respect to the basis $\{e_1, \ldots, e_n\}$.

For a specific example, take $\mathfrak{g} = \mathfrak{so}_3$, the Lie algebra of real 3×3 skew-symmetric matrices and consider the basis of $\mathfrak{g}$ comprising

$$e_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \qquad e_2 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \qquad e_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

with dual e_1^*, e_2^*, e_3^*. Then the structure tensor of $\mathfrak{so}_3$ is

$$\mu_{\mathfrak{so}_3} = \sum_{i,j,k=1}^{3} \varepsilon_{ij}^k e_i^* \otimes e_j^* \otimes e_k,$$

where

$$\varepsilon_{ij}^{k} = \frac{(i-j)(j-k)(k-i)}{2},$$

often called the Levi-Civita symbol.

Example 2 (*Matrix multiplication*) Consider the usual matrix product $\beta : \mathbb{C}^{m\times n} \times \mathbb{C}^{n\times p} \to \mathbb{C}^{m\times p}$, $(A, B) \mapsto AB$. We let E_{ij} be the elementary matrix with one in the (i, j)th entry and zeros elsewhere and E_{ij}^* be the dual. Then the structure tensor for this bilinear operation is

$$\mu_{m,n,p} = \sum_{i,j,k=1}^{m,n,p} E_{ij}^* \otimes E_{jk}^* \otimes E_{ik},$$

the famous *Strassen matrix multiplication tensor*. With respect to these bases, $\mu_{m,n,p}$ is an $mn \times np \times mp$ hypermatrix whose entries are all zeros and ones. When $m = n = p$, this becomes the structure tensor of the matrix algebra $\mathbb{C}^{n\times n}$.

Example 3 (*Matrix–vector multiplication*) Consider the bilinear map

$$\beta : \mathbb{C}^{n\times n} \times \mathbb{C}^n \to \mathbb{C}^n, \quad (A, x) \mapsto Ax.$$

Let $\{E_{ij} \in \mathbb{C}^{n\times n} : i, j = 1, \dots, n\}$ and $\{e_i \in \mathbb{C}^n : i = 1, \dots, n\}$ be the standard bases for $\mathbb{C}^{n\times n}$ and $\mathbb{C}^n$, respectively. Then the structure tensor of β is

$$\mu_\beta = \sum_{i,j=1}^{n} E_{ij}^* \otimes e_j^* \otimes e_i.$$

With respect to these bases, μ_β is an $n^2 \times n \times n$ hypermatrix whose entries are all zeros and ones. This is of course nothing more than a special case of the previous example with $m = n$ and $p = 1$.

A comment is in order for those who are not familiar with multilinear algebra and wonder about the difference between β and μ_β. Given a bilinear map $\beta : U \times V \to W$ there exists a unique trilinear function $\tilde{\beta} : U \times V \times W^* \to \mathbb{C}$ such that given any $(u, v, \omega) \in U \times V \times W^*$ we have

$$\omega(\beta(u, v)) = \tilde{\beta}(u, v, \omega).$$

Furthermore, both β and $\tilde{\beta}$ correspond to the same tensor $\mu_\beta \in U^* \otimes V^* \otimes W$ and so μ_β quantifies both the bilinear operation β and the trilinear operation $\tilde{\beta}$. As a concrete example, consider Example 3 where $\beta : \mathbb{C}^{n\times n} \times \mathbb{C}^n \to \mathbb{C}^n$ is the matrix–vector product

$$\beta(A, x) = Ax.$$

Then $\tilde{\beta} : \mathbb{C}^{n\times n} \times \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}$ is

$$\tilde{\beta}(A, x, y) = y^{\mathsf{T}} A x,$$

and they correspond to the same tensor $\mu_\beta \in (\mathbb{C}^{n\times n})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$. More generally, consider Example 2 where $\beta : \mathbb{C}^{m\times n} \times \mathbb{C}^{n\times p} \to \mathbb{C}^{m\times p}$ is the matrix–matrix product

$$\beta(A, B) = AB.$$

Then $\tilde{\beta} : \mathbb{C}^{m\times n} \times \mathbb{C}^{n\times p} \times \mathbb{C}^{m\times p} \to \mathbb{C}$ is

$$\tilde{\beta}(A, B, C) = \operatorname{tr}(ABC^{\mathsf{T}}),$$

and they correspond to the same tensor $\mu_\beta \in (\mathbb{C}^{m\times n})^* \otimes (\mathbb{C}^{n\times p})^* \otimes \mathbb{C}^{m\times p}$.

We conclude this section with the simplest example, but worked out in full details for the benefit of readers unfamiliar with multilinear algebra.

Example 4 (*Complex number multiplication*) Complex numbers form a two-dimensional algebra over $\mathbb{R}$ and the multiplication of complex numbers is an $\mathbb{R}$-bilinear map

$$\beta : \mathbb{C} \times \mathbb{C} \to \mathbb{C}, \quad (a + bi, c + di) \mapsto (ac - bd) + (ad + bc)i,$$

for any $a, b, c, d \in \mathbb{R}$. Let $e_1 = 1 + 0i = 1$ and $e_2 = 0 + 1i = i$ be the standard basis of $\mathbb{C}$ over $\mathbb{R}$ and let e_1^*, e_2^* be the corresponding dual basis. The structure tensor of $\mathbb{C}$ is, by definition, the structure tensor of β and is given by

$$\mu_{\mathbb{C}} = \mu_\beta = e_1^* \otimes e_1^* \otimes e_1 - e_2^* \otimes e_2^* \otimes e_1 + e_1^* \otimes e_2^* \otimes e_2 + e_2^* \otimes e_1^* \otimes e_2, \tag{2}$$

or, as a hypermatrix with respect to these bases,

$$\mu_{\mathbb{C}} = \left[\begin{array}{cc|cc} 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \end{array}\right] \in \mathbb{R}^{2\times 2\times 2}. \tag{3}$$

We provide here a step-by-step verification that $\mu_{\mathbb{C}}$ is indeed the structure tensor for complex number multiplication over $\mathbb{R}$. Given two complex numbers $z_1 = a+bi$, $z_2 = c + di \in \mathbb{C}$, we write them as $z_1 = ae_1 + be_2$, $z_2 = ce_1 + de_2$. Then

$$\begin{aligned}
\mu_{\mathbb{C}}(z_1, z_2) &= [(e_1^*(z_1)e_1^*(z_2) - e_2^*(z_1)e_2^*(z_2)]e_1 + [e_1^*(z_1)e_2^*(z_2) + e_2^*(z_1)e_1^*(z_2)]e_2 \\
&= [(e_1^*(ae_1 + be_2)e_1^*(ce_1 + de_2) - e_2^*(ae_1 + be_2)e_2^*(ce_1 + de_2)]e_1 \\
&\quad + [e_1^*(ae_1 + be_2)e_2^*(ce_1 + de_2) + e_2^*(ae_1 + be_2)e_1^*(ce_1 + de_2)]e_2 \\
&= (ac - bd)e_1 + (ad + bc)e_2 = (ac - bd, ad + bc) \\
&= (ac - bd) + (ad + bc)i.
\end{aligned}$$

For the uninitiated wondering the usefulness of all these, we will see in Sect. 3 that the notion of tensor rank and its associated rank decomposition allow us to discover

faster, possibly fastest, algorithms for various bilinear operations. For instance, the four-term decomposition in (3) gives us the usual algorithm for multiplying complex numbers but as we will see, one may in fact obtain a three-term decomposition for $\mu_{\mathbb{C}}$,

$$\mu_{\mathbb{C}} = (e_1^* + e_2^*) \otimes (e_1^* + e_2^*) \otimes e_2 + e_1^* \otimes e_1^* \otimes (e_1 - e_2) - e_2^* \otimes e_2^* \otimes (e_1 + e_2). \quad (4)$$

This gives us Gauss's method for multiplying two complex numbers with three real multiplications that we saw in (1). We will have more to say about Gauss's method in Sect. 8—it turns out to be identical to the simplest case of our algorithm for skew-circulant matrix–vector product.

3 Tensor Rank, Border Rank, and Bilinear Complexity

The *Strassen tensor rank method* that we have alluded to in the introduction studies the optimal bilinear complexity of a bilinear operation by studying the rank and border rank [5,28] of its structure tensor.

Definition 1 Let μ_β be the structure tensor of a bilinear map $\beta : U \times V \to W$, we say that the *tensor rank* or just *rank* of μ_β is r if r is the smallest positive integer such that there exist $u_1^*, \dots, u_r^* \in U^*$, $v_1^*, \dots, v_r^* \in V^*$, and $w_1, \dots, w_r \in W$ with

$$\mu_\beta = \sum_{i=1}^{r} u_i^* \otimes v_i^* \otimes w_i. \quad (5)$$

We denote this by $\operatorname{rank}(\mu_\beta) = r$. We say that the *border rank* of μ_β is r if r is the smallest positive integer such that there exists a sequence of tensors $\{\mu_n\}_{n=1}^{\infty}$ of rank r that

$$\lim_{n\to\infty} \mu_n = \mu_\beta.$$

We denote this by $\overline{\operatorname{rank}}\, \mu_\beta = r$. We define the rank and border rank of the zero tensor to be zero.

Our interest in tensor rank is that it gives us exactly *the least number of multiplications* required to evaluate $\beta(u, v)$ for arbitrary inputs u and v. This is established later in Proposition 3. In which case border rank gives the least number of multiplication required to evaluate $\beta(u, v)$ up to arbitrarily high accuracy for arbitrary inputs u and v. The study of complexity of bilinear operations in this manner is called *bilinear complexity*, originally due to Strassen [42,43] and has developed into its own subfield within complexity theory [6, Chapter 14].

To illustrate this, we will start with a simple analysis to show that the usual way of computing matrix–vector product has optimal bilinear complexity, i.e., computing the product of an $m \times n$ matrix and a vector of dimension n requires mn multiplications and one cannot do better.

Proposition 1 *Let $\beta : U \times V \to W$ be a bilinear map and suppose* $\operatorname{span}(\mu_\beta(U \otimes V)) = W$. *Then*

$$\operatorname{rank}(\mu_\beta) \geq \dim W.$$

The role of W may be replaced by U or V.

Proof If not, then

$$\mu_\beta = \sum_{i=1}^{r} u_i^* \otimes v_i^* \otimes w_i \in U^* \otimes V^* \otimes W$$

for some integer $r < \dim W$ and vectors $u_i^* \in U^*$, $v_i^* \in V^*$, $w_i \in W$. Hence

$$\operatorname{span}(\mu_\beta(U \otimes V)) \subsetneq \operatorname{span}\{w_i : i = 1, \ldots, r\}.$$

But this contradicts the assumption that $\mu_\beta(U \otimes V) = W$. □

As an immediate application of Proposition 1, we will show that for a general matrix, the usual way of doing matrix–vector product is already optimal, i.e., Strassen-type fast algorithms for matrix–matrix product do not exist when one of the matrices is a vector (has only one column or row).

Corollary 1 *Let $\beta : \mathbb{C}^{m\times n} \times \mathbb{C}^n \to \mathbb{C}^m$ be the bilinear map defined by the matrix–vector product. Then*

$$\operatorname{rank}(\mu_\beta) = mn.$$

Proof It is easy to see that $\operatorname{rank}(\mu_\beta) \leq mn$. On the other hand,

$$\mu_\beta(\mathbb{C}^n \otimes (\mathbb{C}^m)^*) = (\mathbb{C}^{m\times n})^*,$$

since for the matrix E_{ij} with (i, j)th entry one and zero elsewhere, we have

$$\mu_\beta(e_j \otimes e_i^*)(E_{ij}) = e_i^*(E_{ij}e_j) = 1.$$

Here, $\{e_j : j = 1, \ldots, n\}$ is the standard basis of $\mathbb{C}^n$ and $\{e_i^* : i = 1, \ldots, m\}$ is its dual basis for $(\mathbb{C}^m)^*$. □

This establishes our earlier claim that mn is the minimum number of required multiplications for computing a matrix–vector product. Next we show that we cannot do better than mn even if we are only interested in computing matrix–vector product up to arbitrary accuracy.

Clearly we have

$$\overline{\operatorname{rank}}(\mu_\beta) \leq \operatorname{rank}(\mu_\beta)$$

in general but equality is attained in the following special case. The next proposition turns out to be a very useful result for us—we will rely on it repeatedly to find border ranks of various structure tensors in Sects. 6–17.

Proposition 2 *Let $\beta : U \times V \to W$ be a bilinear map and assume $\mu_\beta(U \otimes V) = W$ and $\operatorname{rank}(\mu_\beta) = \dim W \leq \dim U \dim V$. Then*

$$\overline{\operatorname{rank}}(\mu_\beta) = \operatorname{rank}(\mu_\beta).$$

Proof Assume that $\overline{\operatorname{rank}}(\mu_\beta) < \operatorname{rank}(\mu_\beta)$. Notice that we may regard β as a linear map

$$\beta : U \otimes V \to W.$$

Since $\mu_\beta(U \otimes V) = W$ and $\operatorname{rank}(\mu_\beta) = \dim W$, the rank of β as a linear map (or a matrix) is $\dim W$. Let $r' = \overline{\operatorname{rank}}(\mu_\beta)$. Then there is a sequence $\{\mu_n\}_{n=1}^{\infty}$ of tensors of rank r' such that

$$\lim_{n\to\infty} \mu_n = \mu_\beta.$$

We can similarly regard μ_n as a linear map $\mu_n : U \otimes V \to W$. Since $\operatorname{rank}(\mu_n) = r'$ we see that the rank of μ_n as a linear map is at most r'. By the choice of the sequence $\{\mu_n\}_{n=1}^{\infty}$, we see that

$$\lim_{n\to\infty} \mu_n = \mu_\beta$$

as linear maps (or matrices). Hence, the border rank of μ_β as a linear map (or a matrix) is at most r'. However, for matrices, the notion of rank is the same as border rank. This contradicts the assumption that $\operatorname{rank} \mu_\beta = \dim W > r'$. □

We deduce that the usual way of performing matrix–vector product is also optimal even if we are only interested in approximating the result up to arbitrary accuracy. The following result may also be deduced from the proof (but not the statement) of [37, Lemma 6.1].

Corollary 2 *The border rank of the structure tensor of $m \times n$ matrix–vector product is mn.*

We now give the deferred proof establishing the role of tensor rank in bilinear complexity. This simple result is well-known, classical (see the discussions in [6,41–43]), and has a trivial proof. But given its central importance in our article, we include the statement and proof for easy reference.

Proposition 3 *The rank of μ_β equals the least number of multiplications needed to compute the bilinear map β.*

Proof Given $u \in U$ and $v \in V$, then by definition

$$\mu_\beta(u, v, \cdot) = \beta(u, v) \in W.$$

Since $\operatorname{rank}(\mu_\beta) = r$ we may write μ_β as

$$\mu_\beta = \sum_{i=1}^{r} u_i^* \otimes v_i^* \otimes w_i$$

for some $u_i^* \in U^*$, $v_i^* \in V^*$ and $w_i \in W$. Hence

$$\beta(u, v) = \sum_{i=1}^{r} u_i^*(u) v_i^*(v) w_i \in W.$$

Notice that $u_i^*(u)$ and $v_i^*(v)$ are complex numbers and thus to compute β we only need r multiplications. □

3.1 Remarks on Arithmetic

We highlight a common pitfall in the precise meaning of the word 'multiplication' used in the context of bilinear complexity. Here, it refers strictly to the multiplications of indeterminates but excludes multiplications of a constant and an indeterminate or of two constants. For example, we need one multiplication to calculate $(x, y) \mapsto x \cdot y$ but zero multiplication to calculate $x \mapsto cx$ for any constant $c \in \mathbb{C}$. We will use the term 'scalar multiplication' to refer to the multiplication of two scalars or that of a scalar and an indeterminate over the given field. For instance, $2 \cdot 3$ or $2 \cdot x$ each requires one scalar multiplication to compute.

So in the context of bilinear complexity, a discrete Fourier transform (DFT) may be computed with just $O(1)$, in fact zero, multiplications. The usual $O(n \log n)$ complexity in fast Fourier transform (FFT) counts scalar multiplications. As the case of FFT illustrates, one reason for the exclusion of multiplication involving constants is that this part may often be performed with specialized subroutines or implemented in hardware. On the other hand, bilinear complexity counts only multiplications of variable quantities that could change from input to input.

In particular, traditional studies of structured matrix–vector product, e.g., the superfast algorithms in [36, Chapter 2], rely on the usual measure of computational complexity, counting all arithmetic operations (addition, multiplication, scalar addition, scalar multiplication, etc) as opposed to bilinear complexity. That is why the complexity estimates in [36, Chapter 2] differ substantially from those in Sects. 7–10.

In the most widely studied case of matrix multiplication $\beta : \mathbb{C}^{n\times n} \times \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$, the rank of μ_β informs us about the total arithmetic complexity of β, i.e., counting both additions and multiplications of indeterminates, as the following theorem from [6] shows.

Theorem 2 (Strassen) *Let R_n be the rank of μ_β and let M_n the computational complexity of the matrix multiplication. Then*

$$\inf\{\tau \in \mathbb{R} : M_n = O(n^\tau)\} = \inf\{\tau \in \mathbb{R} : R_n = O(n^\tau)\}.$$

This says that asymptotically, the order of rank of μ_β equals the order of the computational complexity of matrix multiplication. Moreover, combined with Proposition 3, the number of multiplications needed in matrix multiplication dominates the number of additions. This is a very special phenomenon. In general, the number of multiplications cannot dominate the number of additions.

3.2 Remarks on Numerical Stability

Numerical stability has been discussed extensively for Gauss's complex multiplication algorithm in [22], for Strassen–style matrix multiplication algorithms in [16], and for general bilinear operations in [33]. Our goal in this section is to highlight the connection between numerical stability of a bilinear operation β and the nuclear norm [17] of its structure tensor μ_β.

Since numerical stability is an analytic notion, we will need to assume that U^*, V^*, and W are norm spaces. For notational simplicity, we will denote the norms on all three spaces by $\|\cdot\|$. Let $\beta : U \times V \to W$ be a bilinear map and $\mu_\beta \in U^* \otimes V^* \otimes W$ be its structure tensor. We will rewrite the tensor decomposition (5) in the form

$$\mu_\beta = \sum_{i=1}^{r} \lambda_i u_i^* \otimes v_i^* \otimes w_i \tag{6}$$

where $\|u_i^*\| = \|v_i^*\| = \|w_i\| = 1, i = 1, \ldots, r$. As we saw in Proposition 3, any r-term decomposition of the form (6), irrespective of whether r is minimum or not, gives an explicit algorithm for computing β: For any input $u \in U$, $v \in V$, $\beta(u, v) \in W$ is computed as

$$\beta(u, v) = \sum_{i=1}^{r} \lambda_i u_i^*(u) v_i^*(v) w_i.$$

Since the coefficient λ_i captures the increase in magnitude at the ith step, we may regard the sum[2] of (magnitude of) coefficients,

$$\sum_{i=1}^{r} |\lambda_i|, \tag{7}$$

as a measure of the numerical stability of the algorithm corresponding to (6).

[2] This is essential; (7) cannot be replaced by $\left(\sum_{i=1}^{r} |\lambda_i|^p\right)^{1/p}$ for $p > 1$ or $\max_{i=1,\ldots,r} |\lambda_i|$. See [17, Section 3].

As we saw in Proposition 3, when r is minimum, the tensor rank

$$\operatorname{rank}(\mu_\beta) = \min\left\{r : \mu_\beta = \sum\nolimits_{i=1}^{r} \lambda_i u_i \otimes v_i \otimes w_i\right\}$$

gives the least number of multiplications needed to compute β. Analogously, the nuclear norm

$$\|\mu_\beta\|_* = \inf\left\{\sum\nolimits_{i=1}^{r} |\lambda_i| : \mu_\beta = \sum\nolimits_{i=1}^{r} \lambda_i u_i \otimes v_i \otimes w_i,\ r \in \mathbb{N}\right\} \tag{8}$$

quantifies the optimal numerical stability of computing β.

The infimum in (8) is always attained by an r-term decomposition although r may not be $\operatorname{rank}(\mu_\beta)$. For example [17, Proposition 6.1], the structure tensor of complex multiplication in (3) has nuclear norm

$$\|\mu_{\mathbb{C}}\|_* = 4,$$

and is attained by the decomposition (2) corresponding to the usual algorithm for complex multiplication but not the decomposition (4) corresponding to Gauss's algorithm—since the sum of coefficients (upon normalizing the factors) in (4) is $2(1+\sqrt{2})$. In other words, Gauss's algorithm is less stable than the usual algorithm. Nevertheless, in this particular instance, there is a decomposition

$$\mu_{\mathbb{C}} = \frac{4}{3}\left(\left[\frac{\sqrt{3}}{2}e_1 + \frac{1}{2}e_2\right]^{\otimes 3} + \left[-\frac{\sqrt{3}}{2}e_1 + \frac{1}{2}e_2\right]^{\otimes 3} + (-e_2)^{\otimes 3}\right)$$

that attains both $\operatorname{rank}(\mu_{\mathbb{C}})$ and $\|\mu_{\mathbb{C}}\|_*$, i.e., the corresponding algorithm is simultaneously optimal in bilinear complexity and numerical stability.

Numerical stability is a moderately complicated notion [20] and cannot in general be adequately captured by a single number. The sum of coefficients in (7) captures one aspect of numerical stability—it is a measure akin to the *growth factor* in Gaussian elimination with a specific pivoting scheme. The nuclear norm of the structured tensor may then be regarded as an analogue of the minimum growth factor over all possible pivoting strategies.

4 Tensor Ranks of Structure Tensors of Algebras

Let $\mathscr{A}$ be an algebra of dimension n. Let $a_1, \ldots, a_n$ be a basis of $\mathscr{A}$ and $a_1^*, \ldots, a_n^*$ be its dual basis for $\mathscr{A}^*$. Recall that the structure constants c_{ij}^k determine the multiplication operation in $\mathscr{A}$, which we denote by $m_{\mathscr{A}} : \mathscr{A} \times \mathscr{A} \to \mathscr{A}$,

$$m_{\mathscr{A}}(a_i, a_j) = \sum_{k=1}^{n} c_{ij}^k a_k, \quad i, j = 1, \ldots, n.$$

The structure tensor $\mu_{\mathscr{A}} \in \mathscr{A}^* \otimes \mathscr{A}^* \otimes \mathscr{A}$ is then

$$\mu_{\mathscr{A}} = \sum_{i,j,k=1}^{n} c_{ij}^k a_i^* \otimes a_j^* \otimes a_k.$$

Note that $\mu_{\mathscr{A}}$ does not depend on the choice of basis and neither does the tensor and border ranks of $\mu_{\mathscr{A}}$.

When $\mathscr{A} = \mathbb{C}^{n\times n}$, $\mu_{\mathscr{A}} = \mu_{n,n,n}$ is the Strassen matrix multiplication tensor for product of square matrices. Inspired by the Cohn–Umans approach [9] that we will discuss in the next section, we would like to study the rank of $\mu_{\mathscr{A}}$ for an arbitrary algebra, with a view toward embedding an operation whose bilinear complexity is difficult to analyze into an algebra where the task is easier. The first question that we need to answer is the relation between ranks of the respective multiplication tensors. The following proposition appears in [6, Proposition 14.12] but is stated without a proof. While we do not need to use this proposition, we provide a proof that we think is instructive for our tensor rank calculations in Sects. 7–15.

Proposition 4 *If an algebra $\mathscr{A}$ can be embedded into another algebra $\mathscr{B}$, i.e., $\mathscr{A}$ is isomorphic to a subalgebra of $\mathscr{B}$, then* $\operatorname{rank}(\mu_{\mathscr{A}}) \leq \operatorname{rank}(\mu_{\mathscr{B}})$.

Proof Let $j : \mathscr{A} \hookrightarrow \mathscr{B}$ be an embedding of $\mathscr{A}$ into $\mathscr{B}$ as algebras.[3] Then it induces a surjection $j^* : \mathscr{B}^* \twoheadrightarrow \mathscr{A}^*$ and thus a surjection

$$j^* \otimes j^* \otimes \mathrm{id}_{\mathscr{B}} : \mathscr{B}^* \otimes \mathscr{B}^* \otimes \mathscr{B} \twoheadrightarrow \mathscr{A}^* \otimes \mathscr{A}^* \otimes \mathscr{B}.$$

Let $\delta := j^* \otimes j^* \otimes \mathrm{id}_{\mathscr{B}}$. We claim that $\delta(\mu_{\mathscr{B}}) = \mu_{\mathscr{A}}$ and to show this, it suffices to show that

$$\delta(\mu_{\mathscr{B}})(a, a', \cdot) = \mu_{\mathscr{A}}(a, a', \cdot) = m_{\mathscr{A}}(a, a'),$$

which is obvious from the definition of δ. Let r be the rank of $\mu_{\mathscr{B}}$ and suppose $\mu_{\mathscr{B}}$ has a tensor decomposition

$$\mu_{\mathscr{B}} = \sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes b_{i,3}$$

where $b_{i,1}^*, b_{i,2}^* \in \mathscr{B}^*$ and $b_{i,3} \in B$ for $i = 1, \ldots, r$. For notational simplicity, we identify the image of j in $\mathscr{B}$ with $\mathscr{A}$, regarding $\mathscr{A}$ as a subalgebra of $\mathscr{B}$. Let $\mathscr{A}^\perp$ be a subspace of $\mathscr{B}$ so that

$$\mathscr{B} = \mathscr{A} \oplus \mathscr{A}^\perp.$$

[3] Later on in the article we will consider embedding of vector spaces into algebras.

Then we have $b_{i,3} = a_{i,3} + a_{i,3}^{\perp}$ for $i = 1, \ldots, r$ and thus

$$\mu_{\mathscr{B}} = \left(\sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes a_{i,3}\right) + \left(\sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes a_{i,3}^{\perp}\right).$$

Since $\delta(\mu_{\mathscr{B}}) = \mu_{\mathscr{A}}$, we conclude that

$$\delta\left(\sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes a_{i,3}^{\perp}\right) = 0 \quad \text{and} \quad \delta\left(\sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes a_{i,3}\right) = \mu_{\mathscr{A}}.$$

So we obtain an expression of $\mu_{\mathscr{A}}$ as a sum of r rank-one terms,

$$\delta\left(\sum_{i=1}^{r} b_{i,1}^* \otimes b_{i,2}^* \otimes a_{i,3}\right) = \sum_{i=1}^{r} j^*(b_{i,1}^*) \otimes j^*(b_{i,2}^*) \otimes a_{i,3} \in \mathscr{A}^* \otimes \mathscr{A}^* \otimes \mathscr{A},$$

and therefore $\operatorname{rank}(\mu_{\mathscr{A}}) \leq \operatorname{rank}(\mu_{\mathscr{B}})$. □

The map $j^* : \mathscr{B}^* \rightarrowtail \mathscr{A}^*$ above may be viewed as the restriction of linear forms on $\mathscr{B}$ to $\mathscr{A}$.

Corollary 3 *If the algebra $\mathbb{C}^{n\times n}$ can be embedded into an algebra $\mathscr{B}$, then the computational complexity of multiplying two matrices is bounded by the rank of the structure tensor $\mu_{\mathscr{B}}$ of $\mathscr{B}$.*

If we fix bases for $\mathscr{A}$ and $\mathscr{B}$ then we can identify $\mathscr{A}$ with its dual $\mathscr{A}^*$ and $\mathscr{B}$ with its dual $\mathscr{B}^*$. Hence we may regard $\mu_{\mathscr{A}} \in \mathscr{A}^* \otimes \mathscr{A}^* \otimes \mathscr{A}$ as an element of $\mathscr{B}^* \otimes \mathscr{B}^* \otimes \mathscr{B}$. We would like to compare the rank of $\mu_{\mathscr{A}}$ as an element of $\mathscr{A}^* \otimes \mathscr{A}^* \otimes \mathscr{A}$ and as an element of $\mathscr{B}^* \otimes \mathscr{B}^* \otimes \mathscr{B}$. We denote these by $\operatorname{rank}_{\mathscr{A}}(\mu_{\mathscr{A}})$ and $\operatorname{rank}_{\mathscr{B}}(\mu_{\mathscr{A}})$ respectively. We will rely on the following proposition found in [15].

Proposition 5 *Let $U_1, \ldots, U_n$ be vectors spaces and let $U_1', \ldots, U_n'$ be linear subspaces of $U_1, \ldots, U_n$, respectively. Let $T \in U_1' \otimes \cdots \otimes U_n'$. Suppose T has rank r' as an element in $U_1' \otimes \cdots \otimes U_n'$ and has rank r as an element in $U_1 \otimes \cdots \otimes U_n$, then $r = r'$.*

Corollary 4 *If an algebra $\mathscr{A}$ can be embedded into another algebra $\mathscr{B}$, then*

$$\operatorname{rank}_{\mathscr{A}}(\mu_{\mathscr{A}}) = \operatorname{rank}_{\mathscr{B}}(\mu_{\mathscr{A}}).$$

Even though rank stays unchanged under embedding of vector spaces over the same ground field, the same tensor may well have different ranks over different fields. For example [4],

$$T = e_0 \otimes (e_0 \otimes e_0 - e_1 \otimes e_1) + e_1 \otimes (e_0 \otimes e_1 + e_1 \otimes e_0),$$

has rank three (over $\mathbb{R}$) when viewed as an element of $(\mathbb{R}^2)^{\otimes 3}$, but it has rank two (over $\mathbb{C}$) when viewed as an element of $(\mathbb{C}^2)^{\otimes 3}$.

We state here a result of Winograd [26,48], rephrased slightly differently in terms of structure tensors of algebras.

Theorem 3 (Winograd) *Let $p(x)$ be a monic polynomial of degree n whose complete factorization over a given infinite field k is*

$$p(x) = p_1(x)^{e_1} \dots p_q(x)^{e_q}.$$

Then the rank of the structure tensor of the algebra $k[x]/(p(x))$ is $2n - q$.

5 Generalized Cohn–Umans Method and Tensor Rank

A major advance in the study of bilinear complexity of matrix multiplication is the *Cohn–Umans group theoretic method* proposed in [9]. The gist of this idea is that one may compute multiplication in the matrix algebra by 'embedding' it in a judiciously chosen group algebra [9] or, more recently, in an adjacency algebra of a coherent configuration [10]. The embedding in the Cohn–Umans method is, however, not an embedding of algebras as in Sect. 4, but an embedding of vector spaces. As such one needs a certain 'triple product property' to hold to ensure that the entries of the matrix product may still be read off from the entries of the element in the group algebra. In this section, we will generalize the Cohn–Umans method and relate it to tensor rank.

We start by briefly summarizing the Cohn–Umans method. We assume working over $\mathbb{C}$ but the discussions in this and the next section hold for any field. Let G be a finite group and let $\mathbb{C}[G]$ denote its group algebra over $\mathbb{C}$.

Theorem 4 (Cohn–Umans) *Suppose that G contains subsets S, T, U of cardinality m, n, p respectively such that for any $s, s' \in S, t, t' \in T, u, u' \in U, stu = s't'u'$ implies $s = s', t = t', u = u'$. Then for matrices $A = (a_{ij}) \in \mathbb{C}^{m\times n}$, $B = (b_{ij}) \in \mathbb{C}^{n\times p}$ we can associate elements $\widehat{A}, \widehat{B} \in \mathbb{C}[G]$ as follows:*

$$\widehat{A} = \sum_{i,j=1}^{m,n} a_{ij} s_i t_j^{-1}, \quad \widehat{B} = \sum_{i,j=1}^{n,p} b_{ij} t_i u_j^{-1}. \tag{9}$$

The (i, j)th entry of AB is the coefficient of $s_i u_j^{-1}$ in $\widehat{A} \cdot \widehat{B}$.

The condition in the first sentence of the theorem is called the *triple product property*. If such a condition is met, we say that G realizes $\langle m, n, p\rangle$.

The first step toward generalizing the Cohn–Umans method is to view the triple product property in an alternative manner, namely, it is equivalent (see Example 5) to saying that the following diagram commutes:

$$\begin{array}{ccc} \mathbb{C}^{m\times n} \otimes \mathbb{C}^{n\times p} & \xrightarrow{\ j\ } & \mathbb{C}[G] \otimes \mathbb{C}[G] \\ \Big\downarrow{\scriptstyle m} & & \Big\downarrow{\scriptstyle m_G} \\ \mathbb{C}^{m\times p} & \xleftarrow[\text{proj}]{} & \mathbb{C}[G] \end{array} \tag{10}$$

Here j is the embedding of vector spaces defined by (9), proj is the projection map reading off entries of AB from the coefficients of $\widehat{A} \cdot \widehat{B}$, and m and m_G are, respectively, the multiplications of matrices and elements in the group algebra.

To be more precise about the projection map, the translation of matrix multiplication into the multiplication of elements in $\mathbb{C}[G]$ via (9) introduces some 'junk terms', i.e., $\widehat{A} \cdot \widehat{B}$ contains many coefficients that are not needed for obtaining the entries of AB. The projection map is simply a map that picks up the relevant coefficients. As we will see in Sect. 17, there are occasions when that those 'junk terms' could turn out to be useful.

Another important feature of the Cohn–Umans method is that by Wedderburn theorem, $\mathbb{C}[G]$ can be identified with a direct sum of matrix algebras of smaller sizes determined by the irreducible representations of G, giving an efficient way to compute the product $\widehat{A} \cdot \widehat{B}$. Since Wedderburn theorem holds not just for group algebras but for any semisimple algebras, one may in principle use any semisimple algebra $\mathscr{A}$ in place of $\mathbb{C}[G]$. However, motivated by later examples, we will not insist that $\mathscr{A}$ be semisimple—there will be occasions when it is useful to allow $\mathscr{A}$ to have nilpotent elements.

The commutative diagram view of the triple product property (10) allows us to generalize it on an abstract level to arbitrary bilinear operations. Let $\beta : U \times V \to W$ be a bilinear map and let $\mathscr{A}$ be an algebra. If there is an injective linear map $j : U \otimes V \to \mathscr{A} \otimes \mathscr{A}$ and a linear map $\text{proj} : \mathscr{A} \to W$ such that the following diagram commutes:

$$\begin{array}{ccc} U \otimes V & \xrightarrow{\ j\ } & \mathscr{A} \otimes \mathscr{A} \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle m_{\mathscr{A}}} \\ W & \xleftarrow[\text{proj}]{} & \mathscr{A} \end{array} \tag{11}$$

then we can translate the computation of β into multiplication in the algebra $\mathscr{A}$. If this is the case, we will say that the algebra $\mathscr{A}$ *realizes* the bilinear map β.

In case the reader is wondering why $\beta : U \times V \to W$ became $\beta : U \otimes V \to W$ in (11), recall that we do not distinguish between a bilinear map and its corresponding linear map, as we had explained after Proposition 1. We will adopt this convention in the rest of the article without further elaboration.

For readers unfamiliar with such constructions, we used tensor product rather than product in diagrams like (11) because we want to preserve linear structures. If we had used product in (11), then $\beta : U \times V \to W$ has to be a bilinear map, but this does not make sense in the category of vector spaces over $\mathbb{C}$ as morphisms in this category are linear maps (and bilinear maps are in general not linear).

Note that the embedding of $U \otimes V$ into $\mathscr{A} \otimes \mathscr{A}$ and the projection from $\mathscr{A}$ onto W incur zero computational costs in the context of bilinear complexity (no multiplication involved). Hence we obtain

$$\operatorname{rank}(\mu_\beta) \leq \operatorname{rank}(\mu_{\mathscr{A}}).$$

There are occasions (see Sects. 14 and 15) when it is useful to allow a more general framework where the multiplication of the algebra $\mathscr{A}$ is replaced by another bilinear map in (11).

$$\begin{array}{ccc} U \otimes V & \xrightarrow{j} & U' \otimes V' \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle\beta'} \\ W & \xleftarrow[\text{proj}]{} & W' \end{array} \tag{12}$$

For readers familiar with modules [1,29], a further generalization of (12) is to have free modules over a ring in place of vector spaces over a field, i.e.,

$$\begin{array}{ccc} M \otimes_R N & \xrightarrow{j} & M' \otimes_R N' \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle\beta'} \\ L & \xleftarrow[\text{proj}]{} & L' \end{array} \tag{13}$$

where M, N, M', N', L, L' are free modules over a ring R, $\otimes_R$ denotes tensor product of R-modules, and all maps are morphisms of R-modules with j injective.

The commutative diagrams (11), (12), (13) and may be viewed as a *generalized triple product property*. They are similar to [6, Lemma 14.10] but we require j to be injective. Another difference is that we require an embedding of $U \otimes V \to U' \otimes V'$ and this may not necessarily arise from an embedding of $U \times V \to U' \times V'$ as in [6, Lemma 14.10].

We will call the commutative diagrams (11), (12), or (13) *generalized Cohn–Umans method*. We will see several concrete realizations of this abstract framework later. Most of our applications of the generalized Cohn–Umans method will involve (11), but there is one occasion (in Sect. 15) where we need the more general version in (12) and another (in Example 6) where we need (13). In the following, we will establish some existential guarantees for this framework.

Theorem 5 *Let k be a field. Then every bilinear map over k can be realized by an algebra $\mathscr{A}$ over k where the rank of the structure tensor of $\mathscr{A}$ is equal to the rank of the structure tensor of the bilinear map.*

Proof Let $\beta : U \times V \to W$ be a bilinear map and let μ_β be the structure tensor of β. Without loss of generality, we may assume that β is surjective since otherwise we may replace W by the image of β. Assume that $\operatorname{rank}(\mu_\beta) = r$ and that μ_β has a decomposition

$$\mu_\beta = \sum_{i=1}^{r} u_i^* \otimes v_i^* \otimes w_i$$

for some $u_i^* \in U^*$, $v_i^* \in V^*$, $w_i \in W$. Let $\mathscr{A} = k^r$, equipped with entrywise addition and multiplication. Consider the map

$$j : U \times V \to k^r \times k^r$$

defined by $j(u, v) = \big((u_1^*(u), \dots, u_r^*(u), v_1^*(v), \dots, v_r^*(v)\big)$. Lastly define the projection map

$$\operatorname{proj} : k^r \to W, \quad \sum_{i=1}^{r} x_i e_i \mapsto \sum_{i=1}^{r} x_i w_i,$$

where $\{e_i : i = 1, \dots, r\}$ is the standard basis of k^r. It is clear that we have a commutative diagram

$$\begin{array}{ccc} U \otimes V & \xrightarrow{j} & k^r \otimes k^r \\ {\scriptstyle\beta}\downarrow & & \downarrow{\scriptstyle m_{k^r}} \\ W & \xleftarrow[\text{proj}]{} & k^r \end{array}$$

and hence β is realized by the algebra $\mathscr{A} = k^r$. □

Corollary 5 *If k is an algebraically closed field, then every bilinear map over k can be realized by the algebra $k[x]/(x^r - 1)$ where r is the rank of the structure tensor of the bilinear map.*

Proof Apply Theorem 5 and Wedderburn theorem. □

One may argue that Theorem 5 and Corollary 5 are essentially tautological and not particularly useful since the algebra $\mathscr{A}$ involved is commutative (bearing in mind that the Cohn–Umans method becomes uninteresting when the group G is abelian). We will consider another construction that yields a noncommutative algebra. The following construction gives a step-by-step recipe that starts from any given bilinear map β and produces $\mathscr{A}$, a *polynomial identity ring* or *PI ring* [32], for the generalized Cohn–Umans, i.e., (11) is automatically a commutative diagram for this choice of $\mathscr{A}$.

Construction 6 *Let $\{u_1, \dots, u_m\}$ and $\{v_1, \dots, v_n\}$ be bases of U and V. Let $\{u_1^*, \dots, u_m^*\}$ and $\{v_1^*, \dots, v_n^*\}$ be the corresponding dual bases of U^* and V^*. We may assume that β is nondegenerate since otherwise we may simply replace U or V by an appropriate subspace. In which case we may write the structure tensor $\mu_\beta \in U^* \otimes V^* \otimes W$ as*

$$\mu_\beta = \sum_{(i,j) \,:\, \beta(u_i, v_j) \neq 0} u_i^* \otimes v_j^* \otimes w_{ij},$$

i.e., the sum runs over all pairs of (i, j) where $\beta(u_i, v_j) \neq 0$. Let $k\langle x_1, \dots, x_m, y_1, \dots, y_n\rangle$ be the free algebra over k with generators $x_1, \dots, x_m$ and $y_1, \dots, y_n$. We consider an ideal I_0 generated by the relations

$$L(x_i y_j) \sim 0 \quad \textit{if and only if} \quad L(w_{ij}) = 0,$$

where L is a linear form over k in mn variables. Next let $I \supset I_0$ be an ideal of $k\langle x_1, \ldots, x_m, y_1, \ldots, y_n\rangle$ such that I does not contain linear forms in $x_1, \ldots, x_m$ or in $y_1, \ldots, y_n$ and whenever there is a linear form L with $L(x_i y_j) \in I$, $L(x_i y_j) \in I_0$ where at least one (i, j) is a pair of indices such that $w_{ij} \neq 0$ in the representation of μ_β. Then it is easy to verify that the above commutative diagram holds for $\mathscr{A} = k\langle x_1, \ldots, x_m, y_1, \ldots, y_n\rangle/I$, by sending u_i to x_i and v_j to y_j. Such an algebra $\mathscr{A}$ is called a polynomial identity or PI ring.

We will see in Sect. 9 how Construction 6 can be used to obtain the optimum algorithm for Toeplitz matrix–vector product. In principle, it could also be used to obtain the optimum algorithms for Hankel, symmetric, and Hankel-plus-Toeplitz matrix–vector product independently. However, the relations between these structures have allowed us to build upon the algorithms that we obtained earlier and avoid repetitive use of Construction 6.

Example 5 (*Triple product property*) Here $U = \mathbb{C}^{m\times n}$, $V = \mathbb{C}^{n\times p}$, $W = \mathbb{C}^{m\times p}$, we may take $\{E_{ij} : i = 1, \ldots, m;\ j = 1, \ldots, n\}$, $\{E_{ik} : i = 1, \ldots, m; k = 1, \ldots, p\}$ to be the standard basis for U and V. The structure tensor may be expressed as

$$\beta_{m,n,p} = \sum_{i,k=1}^{m,p} \left(\sum_{j=1}^{n} E_{ij}^* \otimes E_{jk}^* \right) \otimes E_{ik}.$$

We see that $\{E_{ik} : i = 1, \ldots, m; k = 1, \ldots, p\}$ is a basis of W and so there is no linear relation among them except the trivial relation $E_{ik} = E_{ik}$. Hence, we obtain an ideal I_0 generated by $x_{ij}y_{jk} - x_{ij'}y_{j'k}$ for all choices of i, j, j', k. As we claim above, find any ideal I such that $I \supset I_0$ does not contain linear forms in x_{ij} or y_{ij}, thus the commutative diagram holds.

Now let us understand the triple product property in this framework. Let G be a group with subsets S, T, U satisfying the triple product property. Then it is obvious that in $k[G]$ there is no nontrivial linear relation for elements of the form $s_i t_j^{-1}$ and $t_j u_k^{-1}$. The triple product property guarantees that $(s_i t_j^{-1})(t_k u_l^{-1}) = (s_i t_j^{-1})(t_k u_l^{-1})$ if and only if $i = i', j = j', k = k'$. Hence, for such G, the commutative diagram holds.

Our construction of $\mathscr{A}$ from the structure tensor is purely formal. Usually, all we could say is that $\mathscr{A}$ is a polynomial identity ring, which does not tell us a lot. However, in special circumstances, when the ideal I is suitably chosen, we can obtain algebras with well-understood properties that we can exploit, as we will see in the rest of this paper.

Before beginning our discussion of structured matrix computations, we give an example to show the broad applicability of the generalized Cohn–Umans method.

Example 6 (*Fast Integer Multiplication*) The algorithms of Karatsuba [25], Toom–Cook [44], [11, pp. 51–77], Schönhage–Strassen [38], Fürer [18], are all instances of (13). In these algorithms, $\mathbb{Z}$ is embedded in the p-adic integers $\mathbb{Z}_p$, and integers are

then multiplied via p-adic multiplication of their images. Consider the commutative diagram of $\mathbb{Z}$-modules

$$\begin{array}{ccc} \mathbb{Z} \otimes_{\mathbb{Z}} \mathbb{Z} & \xrightarrow{j_p} & \mathbb{Z}[x] \otimes_{\mathbb{Z}} \mathbb{Z}[x] \\ \beta \downarrow & & \downarrow \beta' \\ \mathbb{Z} & \xleftarrow[\mathrm{ev}_p]{} & \mathbb{Z}[x] \end{array} \tag{14}$$

where β and β' are multiplications in $\mathbb{Z}$ and $\mathbb{Z}[x]$, respectively. For any $n \in \mathbb{Z}$, we define $f_n(x) := \sum_{i=0}^{d} a_i x^i \in \mathbb{Z}[x]$ where $a_0, \ldots, a_d \in \{0, \ldots, p-1\}$ are such that $n = \sum_{i=0}^{d} a_i p^i$, the base-$p$ (i.e., p-adic) expansion of n. The embedding j_p is defined by

$$j_p(m \otimes n) = f_m(x) \otimes f_n(x),$$

and the evaluation map ev_p sends $f(x) \in \mathbb{Z}[x]$ to $f(p) \in \mathbb{Z}$. Now we may use divide-and-conquer, interpolation, discrete Fourier transform, and fast Fourier transform to multiply the two polynomials, giving us Karatsuba, Toom–Cook, Schönhage–Strassen, and Fürer algorithms respectively.

6 Sparse, Banded, and Triangular Matrices

We begin our study of structured matrices with *sparse matrices*, a particularly simple case that does not require the use of Cohn–Umans method. The results are also unsurprising.

One might wonder what happens to Corollaries 1 and 2 when the matrix involved has zero entries. The answer is what one would expect — the tensor and border rank are both given by the number of nonzero entries. For any $\Omega \subseteq \{1, \ldots, m\} \times \{1, \ldots, n\}$, the set of matrices with *sparsity pattern* Ω is

$$\mathbb{C}^{m \times n}_{\Omega} := \{A \in \mathbb{C}^{m \times n} : a_{ij} = 0 \text{ for all } (i, j) \notin \Omega\},$$

which is clearly a vector space of dimension $\#\Omega$.

Proposition 6 *Let $\beta_\Omega : \mathbb{C}^{m \times n}_{\Omega} \times \mathbb{C}^n \to \mathbb{C}^m, (A, x) \mapsto Ax$. Let $\mu_\Omega \in (\mathbb{C}^{m \times n}_{\Omega})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^m$ be the corresponding structure tensor. Then*

$$\overline{\mathrm{rank}}(\mu_\Omega) = \mathrm{rank}(\mu_\Omega) = \#\Omega.$$

Proof Since the dimension of $\mathbb{C}^{m \times n}_{\Omega}$ is $\#\Omega$ and the usual matrix–vector product costs $\#\Omega$ multiplications, the required result follows from Propositions 1, 2, and 3. □

The set of $n \times n$ *banded matrices* with upper bandwidth k and lower bandwidth l is the special case when

$$\Omega = \{(i, j) \in \{1, \ldots, n\} \times \{1, \ldots, n\} : k < j - i < l\}.$$

Such matrices are called *diagonal* if $(k, l) = 0$, *lower bidiagonal* if $(k, l) = (0, 1)$, *upper bidiagonal* if $(k, l) = (1, 0)$, *tridiagonal* if $(k, l) = (1, 1)$, *pentadiagonal* if $(k, l) = (2, 2)$, *lower triangular* if $(k, l) = (0, n-1)$, *upper triangular* if $(k, l) = (n-1, 0)$.

Corollary 6 *The rank and border rank of the structure tensor of $n \times n$ banded matrix–vector product are both $n + [(n-1) + (n-2) + \cdots + (n-k)] + [(n-1) + (n-2) + \cdots + (n-l)]$.*

Corollary 7 *The rank and border rank of the structure tensor of $n \times n$ upper (or lower) triangular matrix–vector product are both $n(n+1)/2$.*

7 Circulant Matrix

In this section, we consider the problem of computing the product of a circulant matrix with a vector or, equivalently (as we shall see), the product of two circulant matrices. We will obtain an algorithm with optimal bilinear complexity (i.e., minimum number of multiplications). This algorithm will turn out to be similar to the well-known algorithm (e.g. [19, Section 4.8.2] or [36, Section 2.4]) for computing product of two circulant matrices using FFT [12] but we will derive it (i) purely from the perspective of optimal bilinear complexity and (ii) using the Cohn–Umans group theoretic method. However, a key difference is that while the well-known algorithm in [19,36] crucially depends on the use of FFT, our algorithm is indifferent to how DFT is computed—as we have pointed out in Sect. 3.1, DFT incurs no cost in bilinear complexity. This serves as our first example of using the Cohn–Umans method for problems other than matrix multiplication.

We begin by considering a special case of matrix multiplication where we multiply a $1 \times n$ matrix and an $n \times 1$ matrix. Let $a^{\mathsf{T}} = [a_1, \ldots, a_n] \in \mathbb{C}^{1\times n}$ be a row vector and let $b = [b_1, \ldots, b_n]^{\mathsf{T}} \in \mathbb{C}^{n\times 1}$ be a column vector. It is easy to verify that the cyclic group $C_n = \langle g \mid g^n = 1\rangle$ realizes $\langle 1, n, 1\rangle$ via subsets $S = \{1\}, T = C_n, U = \{1\}$ that clearly satisfy the triple product property. By (9), we have

$$\widehat{a} = \sum_{i=1}^{n} a_i g^i, \quad \widehat{b} = \sum_{i=1}^{n} b_i g^{-i}.$$

The coefficients of g^k where $k = 0, 1, \ldots, n-1$ are

$$\sum_{i=1}^{n} a_{i+k} b_i, \quad \text{where } a_s = a_{s'} \text{ iff } s \equiv s' \mod n. \tag{15}$$

To calculate $a^{\mathsf{T}} b$ we just need the coefficient of $1 \in \mathbb{C}[C_n]$ but not the coefficients of the remaining $n-1$ terms (what we called 'junk terms' earlier). On the other hand, if we calculate the product of two circulant matrices, then these $n-1$ terms become useful.

Let $\mathrm{Circ}_n(\mathbb{C})$ be the linear space of all circulant matrices. It is well-known that $\mathrm{Circ}_n(\mathbb{C})$ is closed under the matrix multiplication and so is an algebra.

Proposition 7 *Let $\beta_{\mathrm{c}} : \mathrm{Circ}_n(\mathbb{C}) \times \mathbb{C}^n \to \mathbb{C}^n$, $(\mathrm{Circ}(x), y) \mapsto \mathrm{Circ}(x)y$ be the circulant matrix–vector product and $\mu_{\mathrm{c}} \in \mathrm{Circ}_n(\mathbb{C})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$ be the corresponding structure tensor. Let $\mu_{\mathrm{C}} \in \mathrm{Circ}_n(\mathbb{C})^* \otimes \mathrm{Circ}_n(\mathbb{C})^* \otimes \mathrm{Circ}_n(\mathbb{C})$ be the structure tensor of the algebra $\mathrm{Circ}_n(\mathbb{C})$. Then*

$$\mathrm{rank}(\mu_{\mathrm{c}}) = \mathrm{rank}(\mu_{\mathrm{C}}) = n.$$

Proof A circulant matrix is completely specified by its first column or first row. In particular, since the product of two circulant matrices is still circulant, the product is determined by its first column. Let $x = [x_1, \ldots, x_n]^{\mathsf{T}} \in \mathbb{C}^n$ and let $\mathrm{Circ}(x)$ denote the circulant matrix

$$\mathrm{Circ}(x) = \begin{bmatrix} x_1 & x_2 & \ldots & x_{n-1} & x_n \\ x_n & x_1 & \ldots & x_{n-2} & x_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_3 & x_4 & \ldots & x_1 & x_2 \\ x_2 & x_3 & \ldots & x_n & x_1 \end{bmatrix} \in \mathbb{C}^{n\times n}.$$

Observe that to calculate the matrix–matrix product $\mathrm{Circ}(x)\,\mathrm{Circ}(y)$, it suffices to calculate the matrix–vector product $\mathrm{Circ}(x)[y_1, y_n, \ldots, y_2]^{\mathsf{T}}$. This implies that the structure tensor of the algebra μ_{C} can be obtained from the structure tensor μ_{c} of the bilinear map β_{c} and that

$$\mathrm{rank}(\mu_{\mathrm{c}}) = \mathrm{rank}(\mu_{\mathrm{C}}).$$

To compute $\mathrm{rank}(\mu_{\mathrm{c}})$, observe that for two vectors $x, y \in \mathbb{C}^n$, we have

$$\mathrm{Circ}(x)y = \left[\sum_{i=1}^{n} x_i y_i, \sum_{i=1}^{n} x_i y_{i+1}, \ldots, \sum_{i=1}^{n} x_i y_{i+n-1}\right]^{\mathsf{T}}.$$

Here, we adopt the same convention in (15) that $y_s = y_{s'}$ iff $s \equiv s' \mod n$. Since the entries of $\mathrm{Circ}(x)y$ are exactly the coefficients of the product $\widehat{x} \cdot \widehat{y} \in \mathbb{C}[C_n]$, where $\widehat{x}, \widehat{y} \in \mathbb{C}[C_n]$ are obtained as in (9), it remains to count the number of multiplications needed to form $\widehat{x} \cdot \widehat{y}$ in $\mathbb{C}[C_n]$. Since C_n is the cyclic group of order n, it has exactly n representations, all of dimension one; these are indexed by the roots of unity $1, \omega, \ldots, \omega^{n-1}$ where $\omega = e^{2k\pi \mathrm{i}/n}$. Denote these representations by $V_0, \ldots, V_{n-1}$ where $V_i \simeq \mathbb{C}$ is given by[4]

$$\rho_i : C_n \to \mathbb{C}, \quad g \mapsto \omega^i, \quad i = 0, \ldots, n-1.$$

[4] We do not distinguish between an irreducible representation of G and its irreducible $\mathbb{C}[G]$-submodule.

On the other hand, by Wedderburn theorem we have

$$\mathbb{C}[C_n] \simeq \bigoplus_{i=0}^{n-1} V_i^* \otimes V_i,$$

i.e., we may express elements in $\mathbb{C}[C_n]$ as $n \times n$ diagonal matrices. Explicitly, $\widehat{x} = \sum_{i=0}^{n-1} x_i g^i$ corresponds to the diagonal matrix

$$\operatorname{diag}\left(\sum_{i=0}^{n-1} x_i \omega^i, \sum_{i=0}^{n-1} x_i \omega^{2i}, \ldots, \sum_{i=0}^{n-1} x_i \omega^{(n-1)i}, \sum_{i=0}^{n-1} x_i\right),$$

and $\widehat{y} = \sum_{i=0}^{n-1} y_i g^{-i}$ corresponds to the diagonal matrix

$$\operatorname{diag}\left(\sum_{i=0}^{n-1} y_i \omega^{-i}, \sum y_i \omega^{-2i}, \ldots, \sum_{i=0}^{n-1} y_i \omega^{-(n-1)i}, \sum_{i=0}^{n-1} y_i\right).$$

Therefore, we need n multiplications to compute $\widehat{x} \cdot \widehat{y}$ and thus

$$\operatorname{rank}(\mu_{\mathrm{c}}) \leq n.$$

On the other hand, it follows from Proposition 1 that $\operatorname{rank}(\mu_{\mathrm{C}}) \geq n$ since the image $\mu_{\mathrm{C}}(\operatorname{Circ}_n(\mathbb{C}) \otimes \operatorname{Circ}_n(\mathbb{C}))$ is the whole of $\operatorname{Circ}_n(\mathbb{C})$. □

The proof of Proposition 7 is constructive—it gives an algorithm with optimal bilinear complexity that computes a circulant matrix–vector product or a circulant matrix–circulant matrix product using only n multiplications. In fact, this algorithm is essentially the same as the well-known algorithm for circulant matrix–vector product using FFT.

A departure from usual considerations in numerical linear algebra is that we only care about the number of multiplications used in the algorithm. We minimize the number of multiplications by paying the price of using more additions. We require n^2 additions to execute our algorithm if we have our input $(\operatorname{Circ}(x), y)$ and output $\operatorname{Circ}(x)y$ expressed in the standard basis $e_1, \ldots, e_n$ on $\mathbb{C}^n$. However, if we use the Fourier basis $f_1, \ldots, f_n$ on $\mathbb{C}^n$, i.e., the DFT of $e_1, \ldots, e_n$; then, we require no addition at all to execute our algorithm.

The DFT is a linear map and so computing $f_1, \ldots, f_n$ from $e_1, \ldots, e_n$ would involve only additions and scalar multiplications. Hence, the use of different bases will not change the number of multiplications needed to multiply a circulant matrix to a vector (or two circulant matrices). This agrees with our expectation — a tensor and therefore its rank do not depend on the choice of bases.

Proposition 2 immediately gives us the border rank analogue of Proposition 7. It is also straightforward to obtain the analogue of Proposition 7 for inversion of circulant matrices.

Corollary 8 *The border ranks of the structure tensor of the bilinear operation β_{c} and the structure tensor of the algebra* $\mathrm{Circ}_n(\mathbb{C})$ *are both n, i.e.,*

$$\overline{\mathrm{rank}}(\mu_{\mathrm{c}}) = \overline{\mathrm{rank}}(\mu_{\mathrm{C}}) = n.$$

Corollary 9 *Let $X = \mathrm{Circ}(x)$ be a nonsingular circulant matrix. Then one requires just n divisions to compute its inverse X^{-1}.*

Proof Let $X = \mathrm{Circ}(x)$ where $x = [x_1, \ldots, x_n]^{\mathsf{T}} \in \mathbb{C}^n$ and let $Y = X^{-1}$ be given by $Y = \mathrm{Circ}(y)$ where $y = [y_1, \ldots, y_n]^{\mathsf{T}} \in \mathbb{C}^n$. As in the proof of Proposition 7, their corresponding images in $\mathbb{C}[C_n]$ are

$$\widehat{X} = \mathrm{diag}\left(\sum_{i=0}^{n-1} x_i\omega^i, \sum_{i=0}^{n-1} x_i\omega^{2i}, \ldots, \sum_{i=0}^{n-1} x_i\omega^{(n-1)i}, \sum_{i=0}^{n-1} x_i\right),$$
$$\widehat{Y} = \mathrm{diag}\left(\sum_{i=0}^{n-1} y_i\omega^{-i}, \sum y_i\omega^{-2i}, \ldots, \sum_{i=0}^{n-1} y_i\omega^{-(n-1)i}, \sum_{i=0}^{n-1} y_i\right).$$

Since $\widehat{X} \cdot \widehat{Y} = I$, we obtain

$$\widehat{Y} = \mathrm{diag}\left(\left(\sum_{i=0}^{n-1} x_i\omega^i\right)^{-1}, \left(\sum_{i=0}^{n-1} x_i\omega^{2i}\right)^{-1}, \ldots, \left(\sum_{i=0}^{n-1} x_i\omega^{(n-1)i}\right)^{-1}, \left(\sum_{i=0}^{n-1} x_i\right)^{-1}\right).$$

Hence inverting X requires n divisions.[5] □

8 f-Circulant and Skew-Circulant Matrices

We now extend the work of the previous section to *f-circulant matrices*. For any $f \in \mathbb{C}$, an f-circulant matrix is one of the form

$$\begin{bmatrix} x_1 & x_2 & \ldots & x_{n-1} & x_n \\ fx_n & x_1 & \ldots & x_{n-2} & x_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ fx_3 & fx_4 & \ldots & x_1 & x_2 \\ fx_2 & fx_3 & \ldots & fx_n & x_1 \end{bmatrix} \in \mathbb{C}^{n\times n}.$$

We denote the vector space of $n \times n$ f-circulant matrices by $\mathrm{Circ}_{n,f}(\mathbb{C})$. Evidently, a 1-circulant matrix is just a usual circulant matrix. If $f = -1$, an f-circulant matrix is also called a *skew-circulant matrix*.

[5] The reader is reminded that scalar multiplications by a constant like ω^i are not counted in bilinear complexity.

It is well-known [36, Theorem 2.6.1] and straightforward to see that $\mathrm{Circ}_{n,f}(\mathbb{C}) \simeq \mathbb{C}[x]/(x^n - f)$ and is therefore also an algebra. We may employ the same techniques we used in the case $f = 1$ to prove the following.

Proposition 8 *The rank and border rank of the structure tensor of the f-circulant matrix–vector product over $\mathbb{C}$ are both n. Furthermore, one can invert a nonsingular f-circulant matrix over $\mathbb{C}$ using just n divisions.*

The proofs of these statements are near identical to those in the previous section and we will not repeat them. What we will instead investigate is an interesting special case when $n = 2$ and $f = -1$ but over $\mathbb{R}$ instead of $\mathbb{C}$.

Proposition 9 *The rank of the structure tensor of 2×2 skew-circulant matrix–vector product over $\mathbb{R}$ is three.*

Proof The product of a 2×2 real skew-circulant matrix with a vector is given by

$$X = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \in \mathbb{R}^{2\times 2}, \quad v = \begin{bmatrix} c \\ -d \end{bmatrix} \in \mathbb{R}^2, \quad Xv = \begin{bmatrix} ac - bd \\ -ad - bc \end{bmatrix} \in \mathbb{R}^2.$$

Observe that to compute Xv we require just three real multiplications:

$$M_1 = (a + b)(c + d), \qquad M_2 = ac, \quad M_3 = bd$$

to obtain

$$ac - bd = M_2 - M_3, \quad -ad - bc = -(M_1 - M_2 - M_3). \tag{16}$$

Therefore, the rank r of the structure tensor of 2×2 skew-circulant matrix–vector product over $\mathbb{R}$ is at most three. We show that r cannot be two. Suppose $r = 2$, then there exist polynomials M_1, M_2 in a, b, c, d, each costing only one multiplication to evaluate, such that

$$ac - bd = \alpha_1 M_1 + \alpha_2 M_2, \quad ad + bc = \beta_1 M_1 + \beta_2 M_2.$$

Since a, b, c, d are independent variables, we must also have that

$$\det \begin{bmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{bmatrix} \neq 0.$$

Hence, M_1 and M_2 are both linear combination of $ac - bd$ and $ad + bc$. In particular, there exist $s, t \in \mathbb{R}$ such that

$$M_1 = s(ac - bd) + t(ad + bc).$$

But as M_1 can only involve one multiplication, it must have the form

$$M_1 = (s_1 a + s_2 b + s_3 c + s_4 d)(t_1 a + t_2 b + t_3 c + t_4 d).$$

Without loss of generality we may assume that $s_1 = 0$ and $t_1 = 1$. Then $s_2 = 0$, $s_3 = s$, and $s_4 = t$. These imply that $t_2 = t/s$ and $t_3 = t_4 = 0$ and thus

$$M_1 = (sc + td)\left(a + \frac{t}{s}b\right),$$

giving us $t^2/s = -s$, a contradiction since both s and t are real. □

We show a somewhat unexpected link between Proposition 9 and Example 4.

Corollary 10 *The rank and border rank of $\mu_{\mathbb{C}}$, the structure tensor of $\mathbb{C}$ as an $\mathbb{R}$-algebra, or equivalently, the structure tensor of the $\mathbb{R}$-bilinear map*

$$\beta : \mathbb{C} \times \mathbb{C} \to \mathbb{C}, \quad (a + bi, c + di) \mapsto (ac - bd, ad + bc),$$

are both three.

Proof Let $z_1 = a + bi$ and $z_2 = c + di$. If we identify $\mathbb{C}$ with $\mathbb{R}^2$, then $z_1 z_2 = Xv$ where X and v are as defined in Proposition 9. From which it is clear that the structure tensor $\mu_{\mathbb{C}} \in (\mathbb{R}^2)^* \otimes (\mathbb{R}^2)^* \otimes \mathbb{R}^2$ has rank three over $\mathbb{R}$. The conclusion regarding border rank follows from [15, Theorem 7.1]. □

One may check that the optimal algorithm for skew-circulant matrix–vector product in (16) is in fact the same as Gauss's method for multiplication of complex numbers (1).

9 Toeplitz Matrices

Let $\mathrm{Toep}_n(\mathbb{C})$ be the vector space of $n \times n$ Toeplitz matrices. The following result is well-known, proved in [2] using methods different from those we employ below. Our objective of including this is to provide another illustration of the generalized Cohn–Umans approach where a bilinear operation is embedded in an algebra, in this case, the algebra of circulant matrices $\mathrm{Circ}_{2n}(\mathbb{C})$ in Sect. 7.

Theorem 7 (Bini–Capovani) *Let $\beta_t : \mathrm{Toep}_n(\mathbb{C}) \times \mathbb{C}^n \to \mathbb{C}^n$ be the Toeplitz matrix–vector product. Let $\mu_t \in \mathrm{Toep}_n(\mathbb{C})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$ be the structure tensor of β_t. Then*

$$\mathrm{rank}(\mu_t) = 2n - 1.$$

Proof We begin by observing [14, p. 71] that there is an embedding of an $n \times n$ Toeplitz matrix $X_n = [x_{j-i}]$ as a subblock of a $2n \times 2n$ circulant matrix C_{2n} as follows

$$C_{2n} = \begin{bmatrix} X_n & Y_n \\ Y_n & X_n \end{bmatrix}, \tag{17}$$

where

$$Y_n = \begin{bmatrix} y & x_{-n+1} & x_{-n+2} & \dots & x_{-1} \\ x_{n-1} & y & x_{-n+1} & \dots & x_{-2} \\ x_{n-2} & x_{n-1} & y & \dots & x_{-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & x_3 & \dots & y \end{bmatrix}$$

and $y \in \mathbb{C}$ can be arbitrarily chosen. We will choose

$$y = -\sum_{i=-(n-1)}^{n-1} x_i .$$

and by this choice of y, we only need $2n-1$ multiplications to compute the $2n \times 2n$ circulant matrix–vector product. To see this, recall that in our proof of Proposition 7, the multiplication of Y_n by a vector of dimension $2n$ is computed by multiplying a pair of $2n \times 2n$ diagonal matrices. In our case, the $2n \times 2n$ diagonal matrix corresponding to Y_n is one whose $(1, 1)$th entry is zero. Hence, an $n \times n$ Toeplitz matrix–vector product can be computed with just $2n-1$ multiplications and so the rank of the corresponding structure tensor μ_t is at most $2n-1$. On the other hand, by Proposition 1, $\operatorname{rank}(\mu_t)$ is at least $2n-1$ since $\mu_t(\mathbb{C}^n \otimes (\mathbb{C}^n)^*)$ is the whole of $\operatorname{Toep}_n(\mathbb{C})^*$. □

As in Sect. 7, Proposition 2 and Theorem 7 together give the corresponding result for border rank.

Corollary 11 *The structure tensor of the Toeplitz matrix–vector product has border rank*

$$\overline{\operatorname{rank}}(\mu_t) = 2n - 1.$$

We embedded $\operatorname{Toep}_n(\mathbb{C})$ into $\operatorname{Circ}_{2n}(\mathbb{C}) \simeq \mathbb{C}[C_{2n}]$ and the Toeplitz matrix–vector product inherits a group theoretic interpretation via this embedding. Given $X = [x_{j-i}] \in \operatorname{Toep}_n(\mathbb{C})$ and a vector $z = [z_1, \dots, z_n]^{\mathsf{T}} \in \mathbb{C}^n$, we may explicitly construct the product Xz as follows. First construct two vectors

$$a = [x_0, x_1, \dots, x_{n-1}, y, x_{-n+1}, x_{-n+2}, \dots, x_{-1}]^{\mathsf{T}} \in \mathbb{C}^{2n},$$
$$b = [z_1, \dots, z_n, 0, \dots, 0]^{\mathsf{T}} \in \mathbb{C}^{2n},$$

where $y = -\sum_{i=-(n-1)}^{n-1} x_i$. Notice that

$$a^{\mathsf{T}} b = \sum_{i=1}^{n} x_{i-1} z_i$$

is the first entry of the vector Xz. As we had observed in Sect. 7, the cyclic group $C_{2n} = \langle g \mid g^{2n} = 1 \rangle$ realizes $\langle 1, 2n, 1 \rangle$. Hence, we may construct two elements in $\mathbb{C}[C_{2n}]$, $\widehat{a}$ and $\widehat{b}$ as in Sect. 7 by

$$\widehat{a} = \sum_{i=1}^{n} x_{i-1} g^i + y g^{n+1} + \sum_{i=n+2}^{2n} x_{i-2n-1} g^i \in \mathbb{C}[C_{2n}], \qquad \widehat{b} = \sum_{i=1}^{n} z_i g^{-i} \in \mathbb{C}[C_{2n}].$$

It is easy to see that coefficients of $g^{2n}, g^{2n-1}, \ldots, g^{n+1}$ in $\widehat{a} \cdot \widehat{b}$ give the required entries of Xz.

We have seen in Corollary 9 that inverting a circulant matrix can be done with just n divisions. Although we may embed $\mathrm{Toep}_n(\mathbb{C})$ into $\mathrm{Circ}_{2n}(\mathbb{C})$ via (17) and C_{2n} may be inverted with $2n$ division, there does not seem to be a way to obtain X_n^{-1} from C_{2n}^{-1}.

Suppose we are unaware of the fact that we may embed a Toeplitz matrix into a circulant matrix of larger size, how could we have discovered it? We now provide an illustration of how Construction 6 may be applied systematically to discover the appropriate algebra to use in the generalized Cohn–Umans method for Toeplitz matrix–vector product. Let $\{T_k \in \mathbb{C}^{n\times n} : k = 1, \ldots, 2n-1\}$ be the standard basis for the space of Toeplitz matrices $\mathrm{Toep}_n(\mathbb{C})$, i.e., the entries of $T_k = [t_{ij}]$ are

$$t_{ij} = \begin{cases} 0 & \text{if } j - i \neq k, \\ 1 & \textit{if } j - i = k. \end{cases}$$

Let $\{e_i \in \mathbb{C}^n : i = 1, \ldots, n\}$ be the standard basis of $\mathbb{C}^n$. Then

$$T_k e_i = e_{n-k+i},$$

where $e_i := 0$ whenever $i \geq n+1$ or $i \leq 0$. As described in Sect. 5, we start from the free algebra $\mathbb{C}\langle x_1, \ldots, x_{2n-1}, y_1, \ldots, y_n\rangle$ and let I_0 be the ideal generated by the relations

$$x_k y_i = x_{k'} y_{i'} \quad \text{whenever} \quad 0 \leq k - i = k' - i' \leq n-1,$$

or, equivalently,

$$x_k y_1 = x_{k+i-1} y_i \quad \text{for all} \quad 1 \leq i \leq n,\ 1 \leq k \leq 2n - i. \tag{18}$$

Next we construct an ideal I such that (i) $I_0 \subset I$, (ii) I does not contain any linear form in x_i or y_j, and (iii) whenever $F = L(x_k y_i)$ where L is a linear form and at least one $x_k y_i$ appears in F for some $0 \leq k - i \leq n-1$, then $F \in I_0$. Without loss of generality, we may assume that $y_1 = 1$ and so (18) simplifies to

$$x_k = x_{k+i-1} y_i \quad \text{for all} \quad 1 \leq i \leq n,\ 1 \leq k \leq 2n - i.$$

A moment's thought would then lead us to taking $x_k = x^k$ where x is such that $x^{2n} = 1$, and also $y_i = x^{-i+1} = x^{2n-i+1}$ for $i = 1, \ldots, n$. It is straightforward to check the restrictions we imposed on I are satisfied by these choices, which yield the algebra $\mathbb{C}[x]/(x^{2n} - 1) \simeq \mathbb{C}[C_{2n}]$ that we seek.

We end this section with a brief word on Toeplitz matrix–Toeplitz matrix product $\beta_T : \mathrm{Toep}_n(\mathbb{C}) \times \mathrm{Toep}_n(\mathbb{C}) \to \mathbb{C}^{n\times n}$. Note that $\mathrm{Toep}_n(\mathbb{C})$ is not closed under matrix

multiplication [50]. The corollary below follows from Theorem 7 and the fact that $XY = [Xy_1, \ldots, Xy_n]$ for $X, Y \in \mathrm{Toep}_n(\mathbb{C})$ where y_i is the ith column of Y.

Corollary 12 *The restriction of the matrix multiplication tensor $\mu_{n,n,n}$ to the space $\mathrm{Toep}_n(\mathbb{C})$ of $n \times n$ Toeplitz matrices, regarded as a tensor in $\mathrm{Toep}_n(\mathbb{C})^* \otimes \mathrm{Toep}_n(\mathbb{C})^* \otimes \mathbb{C}^{n\times n}$, has rank at most $n(2n-1)$.*

From the perspective of iterative methods for Toeplitz matrices (both linear systems and least squares), understanding β_t is more important than understanding β_T.

10 Hankel Matrices

The results in this short section follows from those in Sect. 9. However we state them explicitly as these results on Hankel matrices are crucial for those on *symmetric* matrices in Sect. 14, which might come as a surprise.

We introduce a few notations that we will use in Sect. 14. Given a vector $x = [x_0, x_1, \ldots, x_{2n-1}]^{\mathsf{T}} \in \mathbb{C}^{2n}$, we let

$$\mathrm{Hank}(x) := \begin{bmatrix} x_0 & x_1 & \cdots & x_{n-2} & x_{n-1} \\ x_1 & \cdots & \cdots & x_{n-1} & x_n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n-2} & x_{n-1} & \cdots & \cdots & x_{2n-2} \\ x_{n-1} & x_n & \cdots & x_{2n-2} & x_{2n-1} \end{bmatrix} \in \mathbb{C}^{n\times n} \tag{19}$$

be the Hankel matrix defined by x. Let $\mathrm{Hank}_n(\mathbb{C})$ denote the vector space of $n \times n$ Hankel matrices.

The corresponding results for Hankel matrices may be obtained from the ones for Toeplitz matrices essentially via the well-known observation [36, Theorem 2.1.5] that $X \in \mathbb{C}^{n\times n}$ is a Hankel matrix if and only if JX and XJ are both Toeplitz matrices. Here, J is the permutation matrix

$$J := \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{C}^{n\times n}.$$

Since J is a nonsingular linear transformation, μ_h and μ_t must have the same rank and border rank and we obtain the following from Theorem 7, Corollary 11, and Corollary 12.

Corollary 13 *Let $\mu_h \in \mathrm{Hank}_n(\mathbb{C})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$ be the structure tensor of the Hankel matrix–vector product $\beta_h : \mathrm{Hank}_n(\mathbb{C}) \times \mathbb{C}^n \to \mathbb{C}^n$. Then*

$$\mathrm{rank}(\mu_h) = \overline{\mathrm{rank}}(\mu_h) = 2n - 1.$$

Let $\mu_H \in \operatorname{Hank}_n(\mathbb{C})^* \otimes \operatorname{Hank}_n(\mathbb{C})^* \otimes \mathbb{C}^n$ *be the structure tensor of the Hankel matrix–Hankel matrix product* $\beta_H : \operatorname{Hank}_n(\mathbb{C}) \times \operatorname{Hank}_n(\mathbb{C}) \to \mathbb{C}^{n\times n}$. *Then*

$$\operatorname{rank}(\mu_H) \leq n(2n-1).$$

Since $\operatorname{Hank}_n(\mathbb{C}) = J \operatorname{Toep}_n(\mathbb{C}) = \operatorname{Toep}_n(\mathbb{C})J$, one expects a group theoretic realization of the Hankel matrix–vector multiplication. The construction is similar to that of the Toeplitz case.

11 Triangular Toeplitz/Hankel Matrices

We include a discussion of triangular Toeplitz (or Hankel) matrix–vector product because the result may be somewhat unexpected—its optimal bilinear complexity is exactly the same as that of a general Toeplitz (or Hankel) matrix–vector product. The fact that half the entries are zeros cannot be exploited to reduce the number of multiplications in an algorithm. Contrast this with Corollaries 1 and 7. Our methods in this section are new but the results are not, they follow from the work of Bini and Capovani [2].

Let $\operatorname{Toep}_n^{\Delta}(\mathbb{C})$ be the linear space of $n \times n$ upper-triangular Toeplitz matrices and let $\beta_{\Delta} : \operatorname{Toep}_n^{\Delta}(\mathbb{C}) \times \mathbb{C}^n \to \mathbb{C}^n$, $(A, v) \mapsto Av$ denote the upper-triangular Toeplitz matrix–vector product. We claim that the algebra $\mathscr{A} = \mathbb{C}[x]/(x^n)$ realizes β_{Δ}. To see this, let

$$A = \begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \\ 0 & a_0 & \dots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_0 \end{bmatrix} \in \operatorname{Toep}_n^{\Delta}(\mathbb{C}) \quad \text{and} \quad v = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{bmatrix} \in \mathbb{C}^n,$$

we have

$$Av = \begin{bmatrix} a_0v_0 + a_1v_1 + \dots + a_{n-1}v_{n-1} \\ a_0v_1 + \dots + a_{n-2}v_{n-1} \\ \vdots \\ a_0v_{n-1} \end{bmatrix} \in \mathbb{C}^n.$$

Let $A_0, A_1, \dots, A_{n-1}$ be the 'obvious' basis of $\operatorname{Toep}_n^{\Delta}(\mathbb{C})$, i.e., the (i, j)th entry of A_k is one when $j - i = k$ and zero otherwise. Let $e_0, \dots, e_{n-1}$ be the standard basis of $\mathbb{C}^n$. We define an embedding of $\operatorname{Toep}_n^{\Delta}(\mathbb{C}) \otimes \mathbb{C}^n$ into $\mathbb{C}[x]/(x^n) \otimes \mathbb{C}[x]/(x^n)$ taking the bases elements

$$A_i \mapsto x^i, \qquad e_i \mapsto x^{n-1-i}, \qquad i = 0, 1, \dots, n-1.$$

For $A = \sum_{i=0}^{n-1} a_i A_i \in \mathrm{Toep}_n^{\Delta}(\mathbb{C})$ and $v = \sum_{i=0}^{n-1} v_i e_i \in \mathbb{C}^n$, the images $\widehat{A}, \widehat{v} \in \mathbb{C}[x]/(x^n)$ are given by

$$\widehat{A} = a_0 1 + a_1 x + \cdots + a_{n-1} x^{n-1}, \quad \widehat{v} = v_0 x^{n-1} + v_1 x^{n-2} + \cdots + v_{n-1} 1.$$

It is straightforward to verify that $\mathbb{C}[x]/(x^n)$ realizes β_{Δ}. Note that $\mathbb{C}[x]/(x^n)$ is the cohomology ring of the complex projective space $\mathbb{CP}^{n-1}$. In particular it contains nilpotent elements and is not semisimple.

By Theorem 3, the structure tensor of $\mathbb{C}[x]/(x^n)$ has rank $2n-1$, from which we may deduce the following.

Theorem 8 *Let $\mu_{\Delta} \in \mathrm{Toep}_n^{\Delta}(\mathbb{C})^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$ be the structure tensor of the upper-triangular Toeplitz matrix–vector product β_{Δ}. Then*

$$\mathrm{rank}(\mu_{\Delta}) = 2n - 1.$$

Since $\mathrm{Toep}_n^{\Delta}(\mathbb{C})$ is a linear subspace of $\mathrm{Toep}_n(\mathbb{C})$, the structure tensor of upper-triangular Toeplitz matrix–vector product is a projection of the structure tensor of Toeplitz matrix–vector product. However, the tensor ranks of the two structure tensors are both $2n-1$.

12 Toeplitz-Plus-Hankel Matrices

Let S_1 and S_2 be two linear subspaces of $\mathbb{C}^{n\times n}$. Then the set $S_1 + S_2 = \{X_1 + X_2 \in \mathbb{C}^{n\times n} : X_1 \in S_1,\ X_2 \in S_2\}$ is clearly also a linear subspace. If the structure tensors of the matrix–vector product for S_1 and S_2 have ranks r_1 and r_2, respectively, one might guess that the structure tensor of the matrix–vector product for $S_1 + S_2$ has rank $r_1 + r_2$. However, this is not true as we will see below.

Example 7 Let $\mathrm{Toep}_n^{\Delta}(\mathbb{C})$ be the linear subspace of upper-triangular Toeplitz matrices as in Sect. 11. Let $\mathrm{Toep}_n^{\Delta}(\mathbb{C})^{\mathsf{T}}$ be the linear subspace of lower triangualr Toeplitz matrices. Clearly,

$$\mathrm{Toep}_n^{\Delta}(\mathbb{C}) + \mathrm{Toep}_n^{\Delta}(\mathbb{C})^{\mathsf{T}} = \mathrm{Toep}_n(\mathbb{C}).$$

However, by Theorems 7 and 8, the structure tensors of $\mathrm{Toep}_n^{\Delta}(\mathbb{C})$, $\mathrm{Toep}_n^{\Delta}(\mathbb{C})^{\mathsf{T}}$, and $\mathrm{Toep}_n(\mathbb{C})$ all have the same rank $2n-1$.

In the special case $S_1 = \mathrm{Toep}_n(\mathbb{C})$ and $S_2 = \mathrm{Hank}_n(\mathbb{C})$, a matrix in $S_1 + S_2$ is often called a *Toeplitz-plus-Hankel* matrix [34,39]. We show that the value of its rank is one less than the naive guess.

Proposition 10 *The structure tensor of the Toeplitz-plus-Hankel matrix–vector product has rank $4n-3$.*

Proof Let $E \in \mathbb{C}^{n\times n}$ be the matrix of all ones. For any $T \in \mathrm{Toep}_n(\mathbb{C})$ and $H \in \mathrm{Hank}_n(\mathbb{C})$ we have

$$T + H = (T + aE) + (H - aE)$$

and $T + aE \in \mathrm{Toep}_n(\mathbb{C})$, $H - aE \in \mathrm{Hank}_n(\mathbb{C})$ for all $a \in \mathbb{C}$. We show that we may choose an appropriate $a \in \mathbb{C}$ so that the matrix–vector product for $T + aE$ requires only $2n-2$ multiplications. As in the proof of Theorem 7, we may embed $X = T + aE$ into a $2n \times 2n$ circulant matrix

$$C_{2n} = \begin{bmatrix} X & Y \\ Y & X \end{bmatrix}$$

that corresponds to a diagonal matrix whose (1, 1)th entry is zero. We may choose $a \in \mathbb{C}$ so that the (2, 2)th entry of this diagonal matrix is also zero. Hence, the matrix–vector product with $T + aE$ costs at most $2n - 2$ multiplications. Combined with Corollary 13, we see that the structure tensor of the matrix–vector product for $T + H$ has rank at most $4n - 3$. On the other hand, we may check that $\mathrm{Toep}_n(\mathbb{C}) + \mathrm{Hank}_n(\mathbb{C})$ has dimension $4n - 3$. So by Proposition 1, the rank is exactly $4n - 3$. □

13 Block Toeplitz–Toeplitz Block Matrices

One of the most common Toeplitz-like structure in numerical linear algebra is that of a *block Toeplitz–Toeplitz block* or BTTB matrix [7,24,34]. As the name suggests, these are $nk \times nk$ matrices that are $n \times n$ block Toeplitz matrices whose blocks are themselves $k \times k$ Toeplitz matrices, i.e.,

$$A = \begin{bmatrix} X_0 & X_1 & \cdots & X_{n-2} & X_{n-1} \\ X_{-1} & X_0 & \cdots & X_{n-3} & X_{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ X_{2-n} & X_{3-n} & \cdots & X_0 & X_1 \\ X_{1-n} & X_{2-n} & \cdots & X_{-1} & X_0 \end{bmatrix} \in \mathbb{C}^{nk \times nk},$$

where $X_i \in \mathrm{Toep}_k(\mathbb{C})$ for all $i = -(n-1), \ldots, -1, 0, 1, \ldots, n-1$. We write $\mathrm{BTTB}_{n,k}(\mathbb{C})$ for the set of $n \times n$ block Toeplitz matrices with $k \times k$ Toeplitz blocks.

There is of course nothing particularly special about the Toeplitz structure — we may also define block-Hankel-Hankel-block or BHHB matrices, block-circulant-circulant-block or BCCB matrices, etc. In fact we will establish a general result that holds not only for any block matrices with structured blocks but those with *multiple level block structures* (e.g., block Hankel matrices whose blocks are BTTB matrices or block BHHB matrices whose blocks are BCCB matrices).

For each $j = 1, \ldots, s$, let U_{k_j} be a linear subspace of $\mathbb{C}^{k_j \times k_j}$. We define the following linear subspace

$$U_{k_1} \circledast \cdots \circledast U_{k_s} \subseteq \mathbb{C}^{k_1 \times k_1} \circledast \cdots \circledast \mathbb{C}^{k_s \times k_s}$$

where '$\circledast$' denotes the *Kronecker product* [45]. Note that

$$\mathbb{C}^{k_1 \times k_1} \circledast \cdots \circledast \mathbb{C}^{k_s \times k_s} = \mathbb{C}^{k_1 \cdots k_s \times k_1 \cdots k_s}.$$

In particular, the linear subspace of BTTB matrices is obtained by setting $s = 2$, $k_1 = n$, $k_2 = k$, and $U_{k_1} = \mathrm{Toep}_n(\mathbb{C})$, $U_{k_2} = \mathrm{Toep}_k(\mathbb{C})$, i.e.,

$$\mathrm{Toep}_n(\mathbb{C}) \circledast \mathrm{Toep}_k(\mathbb{C}) = \mathrm{BTTB}_{n,k}(\mathbb{C}).$$

For $s = 3$ and $U_{k_i} = \mathrm{Toep}_{k_i}(\mathbb{C})$, $i = 1, 2, 3$, we obtain $k_1 \times k_1$ block Toeplitz matrices whose blocks are $k_2k_3 \times k_2k_3$ BTTB matrices,

$$\mathrm{Toep}_{k_1}(\mathbb{C}) \circledast \mathrm{Toep}_{k_2}(\mathbb{C}) \circledast \mathrm{Toep}_{k_3}(\mathbb{C}) = \mathrm{Toep}_{k_1}(\mathbb{C}) \circledast \mathrm{BTTB}_{k_2,k_3}(\mathbb{C}).$$

Lemma 1 *Let $U \subseteq \mathbb{C}^{n\times n}$ and $V \subseteq \mathbb{C}^{k\times k}$ be linear subspaces. Let*

$$\beta_U : U \times \mathbb{C}^n \to \mathbb{C}^n, \quad \beta_V : V \times \mathbb{C}^k \to \mathbb{C}^k, \quad \beta_{U\circledast V} : (U \circledast V) \times \mathbb{C}^{nk} \to \mathbb{C}^{nk}$$

be the corresponding matrix–vector products with respective structure tensors

$$\mu_U \in U^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n, \quad \mu_V \in V^* \otimes (\mathbb{C}^k)^* \otimes \mathbb{C}^k,$$
$$\mu_{U\circledast V} \in (U \circledast V)^* \otimes (\mathbb{C}^{nk})^* \otimes \mathbb{C}^{nk}.$$

Suppose

$$\mathrm{rank}(\mu_U) = \dim U, \quad \mathrm{rank}(\mu_V) = \dim V \tag{20}$$

and

$$\mu_U(U \otimes \mathbb{C}^n) = \mathbb{C}^n, \quad \mu_V(V \otimes \mathbb{C}^k) = \mathbb{C}^k. \tag{21}$$

Then

$$\mathrm{rank}(\mu_{U\circledast V}) = \mathrm{rank}(\mu_U)\,\mathrm{rank}(\mu_V).$$

Proof It is clear that $\mathrm{rank}(\mu_{U\circledast V})$ is bounded above by $\mathrm{rank}(\mu_U)\,\mathrm{rank}(\mu_V)$. So it suffices to show that $\mathrm{rank}(\mu_{U\circledast V})$ is bounded below by $\mathrm{rank}(\mu_U)\,\mathrm{rank}(\mu_V)$, but this follows from Proposition 1 applied to the matrix–vector product $(U\circledast V)\otimes\mathbb{C}^{nk}\to\mathbb{C}^{nk}$. □

The desired result for BTTB matrices follows immediately from Theorem 7 and Lemma 1.

Corollary 14 *The rank of the structure tensor of the matrix–vector product β_{BTTB} : $\mathrm{BTTB}_{n,k}(\mathbb{C}) \times \mathbb{C}^{nk} \to \mathbb{C}^{nk}$ is $(2k - 1)(2n - 1)$.*

We state a more general version of Lemma 1 that applies to multilevel block structures.

Theorem 9 *For $j = 1, \ldots, s$, let $U_{k_j} \subseteq \mathbb{C}^{k_j\times k_j}$ be a linear subspace of Toeplitz, Hankel, f-circulant, Toeplitz-plus-Hankel, symmetric, or sparse matrices (each U_{k_j} may have a different structure). Let μ_j be the structure tensor of the matrix–vector*

product $U_{k_j} \times \mathbb{C}^{k_j} \to \mathbb{C}^{k_j}$, $j = 1, \ldots, s$, *and let* μ *be that of* $(U_1 \circledast \cdots \circledast U_s) \times \mathbb{C}^{k_1 \cdots k_s} \to \mathbb{C}^{k_1 \cdots k_s}$. *Then*

$$\operatorname{rank}(\mu) = \prod_{j=1}^{s} \operatorname{rank}(\mu_j).$$

Proof By our discussions in the previous and later sections, the conditions (20) and (21) are met for these structured matrices. The result follows by applying Lemma 1 inductively. □

f-circulant matrices include circulant and skew-circulant ones; sparse matrices include banded and triangular ones. Note that we have excluded skew-symmetric matrices and triangular Toeplitz matrices since they do not satisfy (21).

Next we will discuss a Cohn–Umans realization of the matrix–vector product for $U_1 \circledast \cdots \circledast U_s$.

Proposition 11 *If the algebra* $\mathscr{A}_j$ *realizes the bilinear map* $\beta : U_{k_j} \times \mathbb{C}^{k_j} \to \mathbb{C}^{k_j}$ *for* $j = 1, \ldots, s$, *then the tensor product* $\mathscr{A} = \mathscr{A}_1 \otimes \cdots \otimes \mathscr{A}_s$ *realizes the Kronecker product* $U_1 \circledast \cdots \circledast U_s$.

Proof It suffices to prove the statement for $\beta_{U \circledast V} : (U \circledast V) \times \mathbb{C}^{nk} \to \mathbb{C}^{nk}$ when $\beta_U : U \times \mathbb{C}^n \to \mathbb{C}^n$ and $\beta_V : V \times \mathbb{C}^k \to \mathbb{C}^k$ are realized by $\mathscr{A}$ and $\mathscr{B}$ respectively. But this follows from routine arguments: The embeddings $U \hookrightarrow \mathscr{A}$ and $V \hookrightarrow \mathscr{B}$ induce an embedding of $U \circledast V \hookrightarrow \mathscr{A} \otimes \mathscr{B}$ and the projections of $\mathscr{A}$ onto $\mathbb{C}^n$ and $\mathscr{B}$ onto $\mathbb{C}^k$ induce a projection of $\mathscr{A} \otimes \mathscr{B}$ onto $\mathbb{C}^{kn}$. The more general statement then follows from induction. □

For example the matrix–vector product $\beta_{\mathrm{BTTB}} : \mathrm{BTTB}_{n,k}(\mathbb{C}) \times \mathbb{C}^{nk} \to \mathbb{C}^{nk}$ is realized by the algebra

$$\mathscr{A} = \mathbb{C}[x, y]/(x^{2k} - 1,\ y^{2n} - 1),$$

from which we may also deduce the rank of the structure tensor of β_{BTTB}.

14 Symmetric Matrices

We saw in Corollaries 1 and 2 that the usual way of performing matrix–vector product is already optimal for general matrices. A natural question is: What if the matrix is symmetric? This is a very common situation since many, if not most, linear systems and least-squares problems that arise in practice involve symmetric coefficient matrices. Despite this, we are unaware of any previous study. We show here that the optimal bilinear complexity for symmetric matrix–vector product is $n(n+1)/2$. Surprisingly the solution involves Hankel matrices.

We begin with the observation that every symmetric matrix may be expressed as a sum of symmetric Hankel matrices bordered by zeros. A 2×2 symmetric matrix is already a Hankel matrix. The 3×3 and 4×4 cases are shown explicitly below.

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix}, \quad \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix} = \begin{bmatrix} a & b & c \\ b & c & e \\ c & e & f \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & d-c & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\begin{bmatrix} a & b & c & d \\ b & e & f & g \\ c & f & h & i \\ d & g & i & j \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ b & c & d & g \\ c & d & g & i \\ d & g & i & j \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & e-c & f-d & 0 \\ 0 & f-d & e-c & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & h-g-e+c & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The generalization of this observation to $n \times n$ symmetric matrices will be established in our proof below, and together with Corollary 13, be used to deduce the optimal bilinear complexity of symmetric matrix–vector product. Let $\mathrm{S}^2(\mathbb{C}^n)$ be the space of all $n \times n$ symmetric matrices. Let $\beta_s : \mathrm{S}^2(\mathbb{C}^n) \times \mathbb{C}^n \to \mathbb{C}^n$ be the bilinear map of symmetric matrix–vector product and $\mu_s \in \mathrm{S}^2(\mathbb{C}^n)^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n$.

Theorem 10 *The optimal bilinear complexity of symmetric matrix–vector product is* $n(n+1)/2$*, i.e.,* $\operatorname{rank}(\mu_s) = n(n+1)/2$.

Proof By Proposition 1, we see that $\operatorname{rank}(\mu_\beta) \geq \dim \mathrm{S}^2(\mathbb{C}^n) = n(n+1)/2$. On the other hand, for a given

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{1,2} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{1,n-1} & a_{2,n-1} & \dots & a_{n-1,n-1} & a_{n-1,n} \\ a_{1,n} & a_{2,n} & \dots & a_{n-1,n} & a_{n,n} \end{bmatrix} \in \mathrm{S}^2(\mathbb{C}^n)$$

and a column vector $v = [v_1, \dots, v_n]^{\mathsf{T}} \in \mathbb{C}^n$, we claim that $(A, v) \mapsto Av$ can be computed as the sum of several Hankel matrix–vector products of decreasing sizes. Let

$$H_1 = \operatorname{Hank}(a_{1,1}, \dots, a_{1,n}, a_{2,n}, \dots, a_{n,n}) \in \operatorname{Hank}_n(\mathbb{C}),$$

notation as in (19). Then $A - H_1$ is a symmetric matrix of the form

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & A_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \in \mathrm{S}^2(\mathbb{C}^n),$$

where $A_2 \in \mathrm{S}^{(n-2)}(\mathbb{C})$. Also we notice that

$$(A - H_1)v = \begin{bmatrix} 0 \\ A_2 v^{(2)} \\ 0 \end{bmatrix}$$

where $v^{(2)} = [v_2, \ldots, v_{n-1}]^{\mathsf{T}} \in \mathbb{C}^{n-2}$. Now we can repeat the above procedure and inductively we can prove our claim. Explicitly,

$$Av = H_1 v + \begin{bmatrix} 0 \\ H_2 v^{(2)} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ H_3 v^{(3)} \\ 0 \\ 0 \end{bmatrix} + \cdots,$$

where $H_i \in \mathrm{Hank}_{n-2i}(\mathbb{C})$ and $v^{(i)} \in \mathbb{C}^{n-2i}$, $i = 1, \ldots, \lfloor n/2 \rfloor$. H_i and $v^{(i)}$ are linear in the entries of A and v, respectively. By Corollary 13, one can compute $H_i v^{(i)}$ in $2(n-2i)-1$ multiplications. Hence, we obtain

$$\mathrm{rank}(\mu_s) \leq \sum_{i=0}^{\lfloor n/2 \rfloor} [2(n-2i)-1] = \frac{n(n+1)}{2}$$

and therefore $\mathrm{rank}(\mu_s) = n(n+1)/2$. □

One may also interpret the proof of Theorem 10 as an instance of the generalized Cohn–Umans method. We have an embedding of vector spaces

$$\mathrm{S}^2(\mathbb{C}^n)^* \otimes (\mathbb{C}^n)^* \otimes \mathbb{C}^n \hookrightarrow \bigoplus_{i=0}^{\lfloor n/2 \rfloor} \mathrm{Hank}_{n-2i}(\mathbb{C})^* \otimes (\mathbb{C}^{n-2i})^* \otimes \mathbb{C}^{n-2i}, \tag{22}$$

and for each $i = 0, \ldots, \lfloor n/2 \rfloor$, the bilinear map $\mathrm{Hank}_{n-2i}(\mathbb{C}) \times \mathbb{C}^{n-2i} \to \mathbb{C}^{n-2i}$, $(H_i, v^{(i)}) \mapsto H_i v^{(i)}$ is in turn realized by the algebra $\mathbb{C}[x]/(x^{2(n-2i)} - 1)$. Note that the object on the right-hand side of (22) is not an algebra but only a vector space—this is an application of the commutative diagram (12).

15 Skew-Symmetric Matrices

A departure from other sections in this article is that in this section we do not have the optimal bilinear complexity, only upper bounds for it. We first discuss the case of 3×3 skew-symmetric matrix–vector product. Let

$$A = \begin{bmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{bmatrix} \in \Lambda^2(\mathbb{C}^3).$$

Then the usual matrix–vector multiplication gives

$$A \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ay + bz \\ -ax + cz \\ -bx - cy \end{bmatrix},$$

which costs six multiplications. So the rank of the structure tensor of the skew-symmetric matrix–vector product is at most six. We will rely on the following theorem [35,41] for the lower bound of the border rank (hence the rank) of a special 3-tensor.

Theorem 11 *Let $T \in U \otimes V \otimes W$ where $\dim U = \dim V = \dim W = 3$. Let u_1, u_2, u_3 be a basis of U. If we can write T as*

$$T = u_1 \otimes X_1 + u_2 \otimes X_2 + u_3 \otimes X_3,$$

with $X_1, X_2, X_3 \in V \otimes W$ regarded[6] *as 3×3 matrices and if the following block matrix is nonsingular,*

$$M_T = \begin{bmatrix} 0 & X_3 & -X_2 \\ -X_3 & 0 & X_1 \\ X_2 & X_1 & 0 \end{bmatrix} \in \mathbb{C}^{9\times 9},$$

then $\overline{\operatorname{rank}}(T) \geq 5$. The same result holds with V or W in the role of U.

Let $\Lambda^2(\mathbb{C}^n)$ be the space of all $n \times n$ skew-symmetric matrices. Note that $\dim \Lambda^2(\mathbb{C}^3) = 3$ and we may apply Theorem 11 to $T = \mu_\wedge$, the structure tensor of the bilinear map

$$\beta_\wedge : \Lambda^2(\mathbb{C}^3) \times \mathbb{C}^3 \to \mathbb{C}^3, \quad \left(\begin{bmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{bmatrix}, \begin{bmatrix} x \\ y \\ z \end{bmatrix} \right) \mapsto \begin{bmatrix} ay + bz \\ -ax + cz \\ -bx - cy \end{bmatrix}.$$

Let e_1, e_2, e_3 be the standard basis of $\mathbb{C}^3$ and

$$F_1 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad F_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}.$$

be a basis of $\Lambda^2(\mathbb{C}^3)$. Then we may decompose $\mu_\wedge$ as

$$\begin{aligned} \mu_\wedge = {} & F_1 \otimes (e_2 \otimes e_1 - e_1 \otimes e_2) + F_2 \otimes (e_3 \otimes e_1 - e_1 \otimes e_3) \\ & + F_3 \otimes (e_3 \otimes e_2 - e_2 \otimes e_3) \end{aligned}$$

and it is easy to verify that $M_{\mu_\wedge}$ is nonsingular, giving us the following.

Proposition 12 *The rank and border rank of the structure tensor of skew-matrix–vector product for 3×3 matrices are given by*

$$\operatorname{rank}(\mu_\wedge) = 5 \quad \text{or} \quad 6,$$

[6] The result is, however, coordinate independent, i.e., it does not depend on our choice of the bases.

and

$$\overline{\mathrm{rank}}(\mu_\wedge) = 5 \quad or \quad 6.$$

Next we construct an algebra that realizes the 3×3 skew-symmetric matrix–vector product. Our candidate is

$$\mathscr{A} = \mathbb{C}\langle x_1, x_2\rangle / (x_1^2,\ x_2^2,\ x_1x_2 + x_2x_1).$$

The embedding $\Lambda^2(\mathbb{C}^3) \times \mathbb{C}^3 \hookrightarrow \mathscr{A} \times \mathscr{A}$ is given by

$$a_1 \mapsto -x_1, \quad a_2 \mapsto -x_2, \quad a_3 \mapsto 1, \quad b_1 \mapsto 1, \quad b_2 \mapsto -x_2, \quad b_3 \mapsto x_1.$$

Then given $A = \sum_{i=1}^3 u_i a_i \in \Lambda^2(\mathbb{C}^3)$ and $x = \sum_{i=1}^3 v_i b_i \in \mathbb{C}^3$, their images $\widehat{A}, \widehat{x} \in \mathscr{A}$ are given by

$$\widehat{A} = -u_1x_1 - u_2x_2 + u_3, \quad \widehat{x} = v_1 - v_2x_2 + v_3x_1,$$

and their product is given by

$$\begin{aligned}\widehat{A} \cdot \widehat{x} = {} & (-u_1v_1 + u_3v_3) \cdot x_1 + (u_1v_2 + u_2v_3) \cdot x_1x_2 + (-u_2v_1 - u_3v_2) \cdot x_2 \\ & + (u_3v_1) \cdot 1 \in \mathscr{A}.\end{aligned}$$

Hence $\mathscr{A}$ realizes 3×3 skew-symmetric matrix–vector product. We observe that $\mathscr{A}$ may be regarded as the cohomology ring of a torus, i.e., an exterior algebra of a two-dimensional vector space.

We now discuss the general case of $n \times n$ skew-symmetric matrix–vector product $\beta_\wedge : \Lambda^2(\mathbb{C}^n) \times \mathbb{C}^n \to \mathbb{C}^n$. We construct an algebra that realizes $\beta_\wedge$ starting with the inclusion of vector spaces

$$\Lambda^2(\mathbb{C}^n) \hookrightarrow \big(\mathbb{C}[x]/(x^n + 1)\big) \oplus W, \tag{23}$$

where W is a linear subspace of $\mathbb{C}^{n\times n}$ matrices satisfying the following conditions

(i) entries in the first row are all zeros;
(ii) diagonal entries are all zeros;
(iii) entries in the first column satisfy the relation $a_{i,1} + a_{n+2-i,1} = 0$ for $i = 2, \ldots, n$.

Given $A \in \Lambda^2(\mathbb{C}^n)$, the embedding is given by the decomposition

$$A = \begin{bmatrix} 0 & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} \\ -a_{1,2} & 0 & \cdots & a_{2,n-1} & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_{1,n-1} & -a_{2,n-1} & \cdots & 0 & a_{n-1,n} \\ -a_{1,n} & -a_{2,n} & \cdots & -a_{n-1,n} & 0 \end{bmatrix} = A_{\mathrm{c}} + A_{\mathrm{w}},$$

where

$$A_{\mathrm{c}} = \begin{bmatrix} 0 & a_{1,2} & \cdots & a_{1,n-1} & a_{1,n} \\ -a_{1,n} & 0 & \cdots & a_{1,n-2} & a_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_{1,3} & -a_{1,4} & \cdots & 0 & a_{1,2} \\ -a_{1,2} & -a_{1,3} & \cdots & -a_{1,n} & 0 \end{bmatrix} \in \mathrm{Circ}_{n,-1}(\mathbb{C})$$

is a skew-circulant matrix and

$$A_{\mathrm{w}} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ -a_{1,2} + a_{1,n} & 0 & \cdots & a_{2,n-1} - a_{1,n-2} & a_{2,n} - a_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{1,3} - a_{1,n-1} & a_{1,4} - a_{2,n-1} & \cdots & 0 & a_{n-1,n} - a_{1,2} \\ a_{1,2} - a_{1,n} & a_{1,3} - a_{2,n} & \cdots & a_{1,n} - a_{n-1,n} & 0 \end{bmatrix} \in W.$$

Note that A_{c}, being skew-circulant, may be regarded as an element of $\mathbb{C}[x]/(x^n + 1)$ as we have discussed in Sect. 8 and we obtain the embedding in (23).

Since we also have

$$\mathbb{C}^n \simeq \mathbb{C}[x]/(x^n + 1),$$

the bilinear map $\beta_\wedge : \Lambda^2(\mathbb{C}^n) \times \mathbb{C}^n \to \mathbb{C}^n$ may be realized as follows.

$$\begin{array}{ccc}
\Lambda^2(\mathbb{C}^n) \otimes \mathbb{C}^n & \xrightarrow{\quad j \quad} & \big(\mathbb{C}[x]/(x^n+1) \oplus W\big) \otimes \mathbb{C}^n \\
\Big\downarrow{\scriptstyle \mathrm{id}} & & \Big\downarrow{\scriptstyle \simeq} \\
\Lambda^2(\mathbb{C}^n) \otimes \mathbb{C}^n & \xrightarrow{\quad j \quad} & \big(\mathbb{C}[x]/(x^n+1) \otimes \mathbb{C}^n\big) \oplus (W \otimes \mathbb{C}^n) \\
\Big\downarrow{\scriptstyle \mathrm{id}} & & \Big\downarrow{\scriptstyle \simeq} \\
\Lambda^2(\mathbb{C}^n) \otimes \mathbb{C}^n & \xrightarrow{\quad j \quad} & \big(\mathbb{C}[x]/(x^n+1) \otimes \mathbb{C}[x]/(x^n+1)\big) \oplus (W \otimes \mathbb{C}^n) \\
\Big\downarrow{\scriptstyle \beta_\wedge} & & \Big\downarrow{\scriptstyle m} \\
\mathbb{C}^n & \xleftarrow[\mathrm{proj}]{} & \mathbb{C}[x]/(x^n+1) \simeq \mathbb{C}^n
\end{array}$$

We have identified the skew-circulant matrix vector product with the skew-circulant matrix–matrix product (see Sect. 8), i.e., the multiplication

$$\mathbb{C}[x]/(x^n + 1) \times \mathbb{C}^n \to \mathbb{C}^n$$

is identified with the multiplication

$$\mathbb{C}[x]/(x^n + 1) \times \mathbb{C}[x]/(x^n + 1) \to \mathbb{C}[x]/(x^n + 1).$$

This realization is another instance of the commutative diagram (12). To put all these in concrete terms, given $A \in \Lambda^2(\mathbb{C}^n)$ and $x \in \mathbb{C}^n$, we compute the matrix–vector product via

$$Ax = \big(\text{first row of } A_{\mathrm{c}} \operatorname{Circ}(x)\big) + A_{\mathrm{w}} x.$$

Theorem 12 *The rank of the structure tensor of skew-symmetric matrix–vector product is bounded above by* $n^2 - n - \lceil (n-1)/2 \rceil + 1$.

Proof The first factor of the realization is the multiplication in the algebra

$$\mathbb{C}[x]/(x^n + 1) \times \mathbb{C}[x]/(x^n + 1) \to \mathbb{C}[x]/(x^n + 1).$$

By Theorem 3 we see that the rank of the structure tensor of the algebra $\mathbb{C}[x]/(x^n+1)$ is n. The second factor of the realization is a bilinear map

$$W \times \mathbb{C}^n \to \mathbb{C}^n.$$

A matrix–vector product with a matrix in W costs $n^2 - (2n-1) - \lceil (n-1)/2 \rceil$ multiplications — there are $2n-1$ zeros by (i) and (ii) and we invoke Proposition 6; moreover, there are $\lceil (n-1)/2 \rceil$ identical terms by (iii). Therefore, this realization gives an upper bound of $n^2 - n - \lceil (n-1)/2 \rceil + 1$. □

This upper bound is $n + \lceil (n-1)/2 \rceil - 1$ multiplications fewer than the usual matrix–vector product. In particular, for $n = 3$ we obtain the upper bound in Proposition 12.

16 Commutator

Our study of the bilinear complexity of commutators in this section covers only the case of 2×2 matrices. We do not yet know how to extend it to $n \times n$ matrices when $n > 2$.

We consider the bilinear map $[\cdot, \cdot] : \mathbb{C}^{2\times 2} \times \mathbb{C}^{2\times 2} \to \mathbb{C}^{2\times 2}$ defined by $[A, X] = AX - XA$. We will write

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \quad X = \begin{bmatrix} x & y \\ z & w \end{bmatrix},$$

and therefore

$$[A, X] = \begin{bmatrix} bz - cy & (a-d)y - b(x-w) \\ -(a-d)z + c(x-w) & -(bz-cy) \end{bmatrix}.$$

Hence the rank of $\mu_{[\cdot,\cdot]} \in (\mathbb{C}^{2\times 2})^* \otimes (\mathbb{C}^{2\times 2})^* \otimes \mathbb{C}^{2\times 2}$, the structure tensor of $[\cdot, \cdot]$, is at most six.

Now consider the matrix–vector product between matrices and vectors of the following forms

$$\begin{bmatrix} 0 & -c & b \\ -b & a-d & 0 \\ c & 0 & -a-d \end{bmatrix} \in \mathbb{C}^{3\times 3}, \quad \begin{bmatrix} x-w \\ y \\ z \end{bmatrix} \in \mathbb{C}^3.$$

Notice that

$$\begin{bmatrix} 0 & -c & b \\ -b & a-d & 0 \\ c & 0 & -(a-d) \end{bmatrix} \begin{bmatrix} x-w \\ y \\ z \end{bmatrix} = \begin{bmatrix} bz-cy \\ (a-d)y-b(x-w) \\ -(a-d)z+c(x-w) \end{bmatrix}.$$

So the rank and border rank of the structure tensor $\mu_{[\cdot,\cdot]}$ of $[\cdot,\cdot]$ is the same as the rank of the following bilinear operation

$$\beta : \mathbb{C}^3 \times \mathbb{C}^3 \to \mathbb{C}^3, \quad \left(\begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}, \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \right) \mapsto \begin{bmatrix} s_1t_2 + s_2t_3 \\ -s_2t_1 + s_3t_2 \\ -s_1t_1 - s_3t_3 \end{bmatrix}, \tag{24}$$

where

$$s_1 = -c, \quad s_2 = b, \quad s_3 = a-d, \quad t_1 = x-w, \quad t_2 = y, \quad t_3 = z.$$

We will need to distinguish the three copies of $\mathbb{C}^3$ in (24), so for clarity let us denote them by U, V, and W respectively, i.e.,

$$\beta : U \times V \to W.$$

Let $\{u_1, u_2, u_3\}, \{v_1, v_2, v_3\}, \{w_1, w_2, w_3\}$ be the standard bases of U, V, W. Then the structure tensor μ_β of β may be decomposed as

$$\begin{aligned} \mu_\beta = {} & (u_1 \otimes v_2 + u_2 \otimes v_3) \otimes w_1 + (-u_2 \otimes v_1 + u_3 \otimes v_2) \otimes w_2 \\ & + (-u_1 \otimes v_1 - u_3 \otimes v_3) \otimes w_3 \end{aligned}$$

and we may apply Theorem 11 to obtain the following.

Corollary 15 *The rank and border rank of the commutator for* 2×2 *matrices are given by*

$$\operatorname{rank}\big(\mu_{[\cdot,\cdot]}\big) = 5 \quad \text{or} \quad 6$$

and

$$\overline{\operatorname{rank}}\big(\mu_{[\cdot,\cdot]}\big) = 5$$

respectively.

In other words, for $A, X \in \mathbb{C}^{2\times 2}$, computing AX requires at least seven multiplications (e.g., Strassen's algorithm), whereas computing $[A, X] = AX - XA$ requires at most six multiplications. We suspect that this is always the case, i.e., computing commutator is always faster than computing matrix multiplication for $n \times n$ matrices.

We now construct an algebra $\mathscr{A}$ that realizes β and therefore $[\cdot, \cdot]$. Let

$$\mathscr{A} = \mathbb{C}\langle x_1, x_2\rangle/(x_1^2,\ x_2^2,\ x_1x_2 + x_2x_1)$$

and consider the embedding $U \otimes V \to \mathscr{A} \otimes \mathscr{A}$ induced by

$$u_1 \mapsto x_1, \quad u_2 \mapsto x_2, \quad u_3 \mapsto 1, \quad v_1 \mapsto -1, \quad v_2 \mapsto x_2, \quad v_3 \mapsto -x_1.$$

Given $s = \sum_{i=1}^3 s_i u_i \in U$ and $t = \sum_{i=1}^3 t_i v_i \in V$, the images $\widehat{s}$ and $\widehat{t}$ in $\mathscr{A}$ are

$$\widehat{s} = -s_1x_1 - s_2x_2 + s_31, \quad \widehat{t} = -t_11 + t_2x_2 - t_3x_1$$

respectively. Their product is

$$\begin{aligned}\widehat{s}\cdot\widehat{t} &= (s_1x_1 + s_2x_2 + s_31)(-t_11 + t_2x_2 - t_3x_1)\\ &= (-s_1t_1 - s_3t_3)x_1 + (s_1t_2 + s_2t_3)x_1x_2 + (-s_2t_1 + s_3t_2)x_2 + (-s_3t_1)1,\end{aligned}$$

i.e., $\mathscr{A}$ realizes the bilinear map $[\cdot, \cdot]$. Observe that $\mathscr{A}$ is the same algebra that we used to realize the 3×3 skew-symmetric matrix–vector product in Sect. 15.

17 Simultaneous Matrix Multiplication

We round out our list of bilinear operations with two examples of *simultaneous matrix product*.

Proposition 13 *The following two matrix–matrix products:*

$$\begin{bmatrix} a & b\\ c & d\end{bmatrix}\begin{bmatrix} e & f\\ g & h\end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a & b\\ c & d\end{bmatrix}\begin{bmatrix} g & h\\ e & f\end{bmatrix} \tag{25}$$

can be computed simultaneously with eight multiplications.

Proof Let $D_4 = \langle x, y \mid x^4 = y^2 = 1,\ yxy = x^{-1}\rangle$ be the dihedral group of order eight. The multiplication of 2×2 matrices is realized by the subsets

$$H_1 = \langle y\rangle = \{y, 1\}, \quad H_2 = \langle x^2y\rangle = \{x^2y, 1\}, \quad S_3 = \{x^{-1}y, 1\}.$$

Let

$$A = \begin{bmatrix} a & b\\ c & d\end{bmatrix}, \quad B = \begin{bmatrix} e & f\\ g & h\end{bmatrix}. \tag{26}$$

Then A, B correspond to $\widehat{A}, \widehat{B} \in \mathbb{C}[D_4]$ where

$$\begin{aligned}\widehat{A} &= a \cdot (y^{-1}x^2y) + b \cdot (y^{-1}) + c \cdot (x^2y) + d \cdot (1)\\ &= a \cdot (x^2) + b \cdot (y) + c \cdot (x^2y) + d \cdot (1),\\ \widehat{B} &= e \cdot ((x^2y)^{-1}x^{-1}y) + f \cdot ((x^2y)^{-1}) + g \cdot (x^{-1}y) + h \cdot (1)\\ &= e \cdot (x^3) + f \cdot (x^2y) + g \cdot (x^3y) + h \cdot (1).\end{aligned}$$

We compute the product $\widehat{A} \cdot \widehat{B}$ in $\mathbb{C}[D_4]$,

$$\begin{aligned}\widehat{A} \cdot \widehat{B} = {} & (ae + bg) \cdot (x) + (af + bh) \cdot (y) + (ce + dg) \cdot (x^3y) + (cf + dh) \cdot (1)\\ & + (ag + be) \cdot (xy) + (ah + bf) \cdot (x^2)\\ & + (cg + de) \cdot (x^3) + (ch + df) \cdot (x^2y)\end{aligned}$$

and observe that the first four terms and last four terms are precisely the entries of

$$M_1 := \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} e & f \\ g & h \end{bmatrix} \quad \text{and} \quad M_2 := \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} g & h \\ e & f \end{bmatrix}$$

respectively. In other words, we can calculate M_1 and M_2 simultaneously by calculating $\widehat{A} \cdot \widehat{B}$. On the other hand, D_4 has four irreducible representations of dimension one and one of dimension two:

(i) trivial: $(1, x, x^2, x^3, y, xy, x^2y, x^3y) \overset{\rho}{\mapsto} (1, 1, 1, 1, 1, 1, 1, 1)$
(ii) sign type 1 : $(1, x, x^2, x^3, y, xy, x^2y, x^3y) \overset{\rho}{\mapsto} (1, 1, 1, 1, -1, -1, -1, -1)$
(iii) sign type 2 : $(1, x, x^2, x^3, y, xy, x^2y, x^3y) \overset{\rho}{\mapsto} (1, -1, 1, -1, 1, -1, 1, -1)$
(iv) sign type 3 : $(1, x, x^2, x^3, y, xy, x^2y, x^3y) \overset{\rho}{\mapsto} (1, -1, 1, -1, -1, 1, -1, 1)$
(v) two-dimensional:

$$1 \mapsto \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad x \mapsto \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad x^2 \mapsto \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad x^3 \mapsto \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$
$$y \mapsto \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad xy \mapsto \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad x^2y \mapsto \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad x^3y \mapsto \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}.$$

By Wedderburn theorem we have

$$\mathbb{C}[D_4] \simeq \mathbb{C} \oplus \mathbb{C} \oplus \mathbb{C} \oplus \mathbb{C} \oplus \mathbb{C}^2 \otimes \mathbb{C}^2,$$

where the first four $\mathbb{C}$'s correspond to the four 1-dimensional representations and the $\mathbb{C}^2 \otimes \mathbb{C}^2$ corresponds to the 2-dimensional representation. Under this isomorphism, we may identify $\widehat{A}$ and $\widehat{B}$ as 6×6 block diagonal matrices,

$$\widehat{A} = \begin{bmatrix} a+b+c+d & 0 & 0 & 0 & 0 & 0 \\ 0 & a-b-c+d & 0 & 0 & 0 & 0 \\ 0 & 0 & a+b+c+d & 0 & 0 & 0 \\ 0 & 0 & 0 & a-b-c+d & 0 & 0 \\ 0 & 0 & 0 & 0 & -a+b-c+d & 0 \\ 0 & 0 & 0 & 0 & 0 & -a-b+c+d \end{bmatrix}$$

and

$$\widehat{B} = \begin{bmatrix} e+f+g+h & 0 & 0 & 0 & 0 & 0 \\ 0 & e-f-g+h & 0 & 0 & 0 & 0 \\ 0 & 0 & -e+f-g+h & 0 & 0 & 0 \\ 0 & 0 & 0 & -e-f+g+h & 0 & 0 \\ 0 & 0 & 0 & 0 & -f+h & e-g \\ 0 & 0 & 0 & 0 & -e-g & f+h \end{bmatrix}.$$

Hence the computation of $\widehat{A} \cdot \widehat{B}$ costs eight multiplications. □

Corollary 16 *Suppose*

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} g & h \\ e & f \end{bmatrix} = 0.$$

Then the product

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix}$$

can be computed with four multiplications.

Proof If the given condition holds, one may obtain the required product from the first four diagonal entries of $\widehat{A} \cdot \widehat{B}$, which costs four multiplications. □

We restate Proposition 13 in terms of the structure tensor μ_{f} of the bilinear map

$$\beta_{\mathsf{f}} : \mathbb{C}^{2\times 2} \times \mathbb{C}^{2\times 2} \to \mathbb{C}^{2\times 2} \oplus \mathbb{C}^{2\times 2}, \quad (A, B) \mapsto (AB, AB^{\mathsf{f}}),$$

where B^{f} denotes the operation of switching the first and second row of B. Note that

$$\mu_{\mathsf{f}} \in (\mathbb{C}^{2\times 2})^* \otimes (\mathbb{C}^{2\times 2})^* \otimes (\mathbb{C}^{2\times 2} \oplus \mathbb{C}^{2\times 2}) \simeq \mathbb{C}^4 \otimes \mathbb{C}^4 \otimes \mathbb{C}^8.$$

Proposition 14 *The rank and border rank of the structure tensor for the simultaneous matrix–matrix product in* (25) *are given by*

$$\operatorname{rank}(\mu_{\mathsf{f}}) = \overline{\operatorname{rank}}(\mu_{\mathsf{f}}) = 8.$$

Proof It is easy to verify that $\operatorname{span}(\mu_{\mathsf{f}}(\mathbb{C}^{2\times 2} \otimes \mathbb{C}^{2\times 2})) = \mathbb{C}^{2\times 2}$. Hence, the required result follows from Propositions 1, 2, and 13. □

Corollary 17 *Consider the matrices*

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathbb{C}^{2\times 2}, \quad B = \begin{bmatrix} e_1 & e_2 & \cdots & e_{2n} \\ f_1 & f_2 & \cdots & f_{2n} \end{bmatrix} \in \mathbb{C}^{2\times 2n}, \quad B^{\mathsf{f}} = \begin{bmatrix} f_1 & f_2 & \cdots & f_{2n} \\ e_1 & e_2 & \cdots & e_{2n} \end{bmatrix} \in \mathbb{C}^{2\times 2n}, \tag{27}$$

where n is any positive integer. Then AB and AB^{f} can be computed simultaneously with $8n$ multiplications.

Proof We may realize the bilinear map

$$\mathbb{C}^{2\times 2} \times \mathbb{C}^{2\times 2n} \to \mathbb{C}^{2\times 2n} \oplus \mathbb{C}^{2\times 2n}, \quad (A, B) \mapsto (AB, AB^{\mathsf{f}})$$

by the algebra $\mathbb{C}[D_4] \times \cdots \times \mathbb{C}[D_4]$ (n copies). □

Suppose we are instead interested in computing

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} e & f \\ g & h \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} h & g \\ e & f \end{bmatrix}$$

simultaneously. We start by realizing 2×2 matrix product by the algebra $\mathbb{C}[x]/(x^8-1)$. Let $A, B \in \mathbb{C}^{2\times 2}$ be as in (26). Consider the embedding

$$\begin{aligned} j : \mathbb{C}^{2\times 2} \otimes \mathbb{C}^{2\times 2} &\to \mathbb{C}[x]/(x^8-1) \otimes \mathbb{C}[x]/(x^8-1), \\ (A, B) &\mapsto (ax^3 + cx^2 + bx + d, gx^6 + ex^4 + hx^2 + f), \end{aligned}$$

and the projection

$$\operatorname{proj} : \mathbb{C}[x]/(x^8-1) \to \mathbb{C}^{2\times 2}, \quad \sum_{i=1}^{7} u_i x^i \mapsto \begin{bmatrix} u_7 & u_3 \\ u_6 & u_2 \end{bmatrix}.$$

We may verify that for these choices, the diagram in (11) commutes. The product

$$(ax^3 + cx^2 + bx + d)(gx^6 + ex^4 + hx^2 + f)$$

in $\mathbb{C}[x]/(x^8-1)$ gives us the following counterpart of Proposition 13.

Proposition 15 *The following two matrix–matrix products:*

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} e & f \\ g & h \end{bmatrix} \quad \textit{and} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} h & g \\ e & f \end{bmatrix} \tag{28}$$

can be computed simultaneously with eight multiplications.

Again, we may restate Proposition 15 in terms of the structure tensor μ_{g} of the bilinear map

$$\beta_{\mathsf{g}} : \mathbb{C}^{2\times 2} \times \mathbb{C}^{2\times 2} \to \mathbb{C}^{2\times 2} \oplus \mathbb{C}^{2\times 2}, \quad (A, B) \mapsto (AB, AB^{\mathsf{g}})$$

where B^{g} is the matrix obtained from B by switching the first row and the second row and then switching the first and the second entry in the first row. The following analogue of Proposition 14 follows from Propositions 1, 2, and 15.

Proposition 16 *The rank and border rank of the structure tensor for the simultaneous matrix–matrix product in* (28) *are given by*

$$\operatorname{rank}(\mu_{\mathsf{g}}) = \overline{\operatorname{rank}}(\mu_{\mathsf{g}}) = 8.$$

We also have the following analogue of Corollary 17.

Corollary 18 *Let $A \in \mathbb{C}^{2\times 2}$ and $B \in \mathbb{C}^{2\times 2n}$ be as in (27). Let $B^{\mathfrak{g}} \in \mathbb{C}^{2\times 2n}$ be the matrix obtained from B by switching the first and second row followed by switching $2i$th and $(2i-1)$th entry in the first row for $i = 1, 2, \ldots, \lfloor n/2 \rfloor$. Then AB and $AB^{\mathfrak{g}}$ can be computed simultaneously with $8n$ multiplications.*

Proof We may realize the bilinear map

$$\mathbb{C}^{2\times 2} \times \mathbb{C}^{2\times 2n} \to \mathbb{C}^{2\times 2n} \oplus \mathbb{C}^{2\times 2n}, \quad (A, B) \mapsto (AB, AB^{\mathfrak{g}})$$

by the algebra $\mathbb{C}[x_1, \ldots, x_n]/(x_i^8 - 1 \mid i = 1, \ldots, n)$. □

18 Conclusion

The Strassen tensor rank approach gives us a simple way for quantifying bilinear complexity, whereas the (generalized) Cohn–Umans approach gives us a constructive way that allows the rich properties of various algebras to be used in analyzing bilinear complexity. The two methods can be applied hand-in-hand to systematically discover algorithms of optimal bilinear complexity.

Acknowledgements We thank Henry Cohn for very helpful discussions that initiated this work. We are also grateful to Andrew Chien, Nikos Pitsianis, and Xiaobai Sun for answering our questions about energy costs and circuit complexity of various integer and floating point operations; to Mike Stein for suggesting that we examine BTTB matrices; and to Chris Umans for prompting Construction 6. We thank the two anonymous referees and the handling editor for their exceptionally helpful comments and constructive suggestions. In particular, we included Sects. 1.2 and 3.2 at the handling editor's urging, which in retrospect were glaring omissions. LHL and KY are partially supported by AFOSR FA9550-13-1-0133, DARPA D15AP00109, NSF IIS 1546413, DMS 1209136, and DMS 1057064. In addition, KY's work is also partially supported by NSF CCF 1017760.

References

1. W. A. Adkins and S. H. Weintraub, *Algebra: An approach via module theory*, Graduate Texts in Mathematics, **136**, Springer, New York, 1992.
2. D. Bini and M. Capovani, "Tensor rank and border rank of band Toeplitz matrices," *SIAM J. Comput.*, **16** (1987), no. 2, pp. 252–258.
3. Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA, 1996.
4. J.-L. Brylinski, "Algebraic measures of entanglement," pp. 3–23, G. Chen and R. K. Brylinski (Eds), *Mathematics of Quantum Computation*, CRC, Boca Raton, FL, 2002.
5. J. Buczyński and J. M. Landsberg, "Ranks of tensors and a generalization of secant varieties," *Linear Algebra Appl.*, **438** (2013), no. 2, pp. 668–689.
6. P. Bürgisser, M. Clausen, and M. A. Shokrollahi, *Algebraic Complexity Theory*, Grundlehren der Mathematischen Wissenschaften, **315**, Springer-Verlag, Berlin, 1997.
7. R. H.-F. Chan and X.-Q. Jin, *An Introduction to Iterative Toeplitz Solvers*, Fundamentals of Algorithms, **5**, SIAM, Philadelphia, PA, 2007.
8. H. Cohn, R. Kleinberg, B. Szegedy, and C. Umans, "Group-theoretic algorithms for matrix multiplication," *Proc. IEEE Symp. Found. Comput. Sci.* (FOCS), **46** (2005), pp. 379–388.
9. H. Cohn and C. Umans, "A group-theoretic approach to fast matrix multiplication," *Proc. IEEE Symp. Found. Comput. Sci.* (FOCS), **44** (2003), pp. 438–449.
10. H. Cohn and C. Umans, "Fast matrix multiplication using coherent configurations," *Proc. ACM–SIAM Symp. Discrete Algorithms* (SODA), **24** (2013), pp. 1074–1087.

11. S. A. Cook, *On the Minimum Computation Time of Functions*, Ph.D. thesis, Harvard University, Cambridge, MA, 1966.
12. J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comp.*, **19** (1965), no. 90, pp. 297–301.
13. D. Coppersmith and S. Winograd, "Matrix multiplication via arithmetic progressions," *J. Symbolic Comput.*, **9** (1990), no. 3, pp. 251–280.
14. P. J. Davis, *Circulant Matrices*, John Wiley, New York, NY, 1979.
15. V. De Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Anal. Appl.*, **30** (2008), no. 3, pp. 1084–1127.
16. J. Demmel, I. Dumitriu, O. Holtz, and R. Kleinberg, "Fast matrix multiplication is stable," *Numer. Math.*, **106** (2007), no. 2, pp. 199–224.
17. S. Friedland and L.-H. Lim, "Nuclear norm of higher-order tensors," (2016). http://arxiv.org/abs/1410.6072.
18. M. Fürer, "Faster integer multiplication," *SIAM J. Comput.*, **39** (2009), no. 3, pp. 979–1005.
19. G. Golub and C. Van Loan, *Matrix Computations*, 4th Ed., Johns Hopkins University Press, Baltimore, MD, 2013.
20. N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2nd Ed., SIAM, Philadelphia, PA, 2002.
21. N. J. Higham, *Functions of Matrices*, SIAM, Philadelphia, PA, 2008.
22. N. J. Higham, "Stability of a method for multiplying complex matrices with three real matrix multiplications," *SIAM J. Matrix Anal. Appl.*, **13** (1992), no. 3, pp. 681–687.
23. *Intel 64 and IA-32 Architectures Optimization Reference Manual*, September 2015. http://www.intel.com/content/dam/www/public/us/en/documents/manuals/64-ia-32-architectures-optimization-manual
24. T. Kailath and J. Chun, "Generalized displacement structure for block-Toeplitz, Toeplitz-block, and Toeplitz-derived matrices," *SIAM J. Matrix Anal. Appl.*, **15** (1994), no. 1, pp. 114–128.
25. A. Karatsuba and Yu. Ofman, "Multiplication of many-digital numbers by automatic computers," *Dokl. Akad. Nauk SSSR*, **145** (1962), pp. 293–294 [English translation: *Soviet Phys. Dokl.*, **7** (1963), pp. 595–596].
26. D. E. Knuth, *The Art of Computer Programming*, Volume 2: Seminumerical algorithms, 3rd Ed., Addison–Wesley, Reading, MA, 1998.
27. V. K. Kodavalla, "IP gate count estimation methodology during micro-architecture phase," *IP Based Electronic System Conference and Exhibition* (IP-SOC), Grenoble, France, December 2007. http://www.design-reuse.com/articles/19171/ip-gate-count-estimation-micro-architecture-phase.html
28. J. M. Landsberg, *Tensors: Geometry and Applications*, Graduate Studies in Mathematics, **128**, AMS, Providence, RI, 2012.
29. S. Lang, *Algebra*, Rev. 3rd Ed., Graduate Texts in Mathematics, **211**, Springer, New York, NY, 2002.
30. F. Le Gall, "Powers of tensors and fast matrix multiplication," *Proc. Internat. Symp. Symbolic Algebr. Comput.* (ISSAC), **39** (2014), pp. 296–303.
31. L.-H. Lim, "Tensors and hypermatrices," in: L. Hogben (Ed.), *Handbook of Linear Algebra*, 2nd Ed., CRC Press, Boca Raton, FL, 2013.
32. J. C. McConnell and J. C. Robson, *Noncommutative Noetherian Rings*, Rev. Ed., Graduate Studies in Mathematics, **30**, AMS, Providence, RI, 2001.
33. W. Miller, "Computational complexity and numerical stability," *SIAM J. Comput.*, **4** (1975), no. 2, pp. 97–107.
34. M. K. Ng, *Iterative Methods for Toeplitz Systems*, Oxford University Press, New York, NY, 2004.
35. G. Ottaviani, "Symplectic bundles on the plane, secant varieties and Lüroth quartics revisited," *Quad. Mat.*, **21** (2007), pp. 315–352.
36. V. Y. Pan, *Structured Matrices and Polynomials: Unified superfast algorithms*, Birkhäuser, Boston, MA, 2001.
37. A. Schönhage, "Partial and total matrix multiplication," *SIAM J. Comput.*, **10** (1981), no. 3, pp. 434–455.
38. A. Schönhage and V. Strassen, "Schnelle Multiplikation großer Zahlen," *Computing*, **7** (1971), no. 3, pp. 281–292.
39. G. Strang and S. MacNamara, "Functions of difference matrices are Toeplitz plus Hankel," *SIAM Rev.*, **56** (2014), no. 3, pp. 525–546.
40. V. Strassen, "Gaussian elimination is not optimal," *Numer. Math.*, **13** (1969), no. 4, pp. 354–356.

41. V. Strassen, "Rank and optimal computation of generic tensors," *Linear Algebra Appl.*, **52/53** (1983), pp. 645–685.
42. V. Strassen, "Relative bilinear complexity and matrix multiplication," *J. Reine Angew. Math.*, **375/376** (1987), pp. 406–443.
43. V. Strassen, "Vermeidung von Divisionen," *J. Reine Angew. Math.*, **264** (1973), pp. 184–202.
44. A. L. Toom, "The complexity of a scheme of functional elements realizing the multiplication of integers," *Dokl. Akad. Nauk SSSR*, **150** (1963), pp. 496–498 [English translation: *Soviet Math. Dokl.*, **4** (1963), pp. 714–716].
45. C. F. Van Loan, "The ubiquitous Kronecker product," *J. Comput. Appl. Math.*, **123** (2000), no. 1–2, pp. 85–100.
46. V. Vassilevska Williams, "Multiplying matrices faster than Coppersmith–Winograd," *Proc. ACM Symp. Theory Comput.* (STOC), **44** (2012), pp. 887–898.
47. D. S. Watkins, *The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods*, SIAM, Philadelphia, PA, 2007.
48. S. Winograd, "Some bilinear forms whose multiplicative complexity depends on the field of constants," *Math. Syst. Theory*, **10** (1976/77), no. 2, pp. 169–180.
49. K. Ye and L.-H. Lim, "Algorithms for structured matrix-vector product of optimal bilinear complexity," *Proc. IEEE Inform. Theory Workshop* (ITW), **16** (2016), to appear.
50. K. Ye and L.-H. Lim, "Every matrix is a product of Toeplitz matrices," *Found. Comput. Math.*, **16** (2016), no. 3, pp. 577–598.