$O(n^3)$ Diagonalization over $\mathbb{F}_p$ via the $A^p = A$ criterion

HyeongRae Jo
bluestaxks@icloud.com
github.com/BlueStaxks

2024/1/20

Abstract

This paper presents a simple criterion and a practical algorithm for diagonalizing matrices over a prime field $\mathbb{F}_p$. First, I prove that $A \in \mathbb{F}_p^{n\times n}$ is diagonalizable over $\mathbb{F}_p$ if and only if $A^p = A$; for invertible A this is equivalent to $A^{p-1} = I$. The proof is short via the minimal-polynomial criterion and the fact that $x^p - x$ is square-free in $\mathbb{F}_p[x]$. Second, I develop a diagonalization method that avoids characteristic/minimal polynomials and any polynomial factoring. The procedure runs in $O(n^3)$ time for fixed p, and exposes fine-grained parallelism suitable for GPUs.

Theorem and Minimal-Polynomial Proof

We work over $\mathbb{F}_p$ with p prime. We will use two very standard facts:

(F1) **Frobenius over $\mathbb{F}_p$.** For every $a \in \mathbb{F}_p$, $a^p = a$. Hence

$$x^p - x = \prod_{a\in\mathbb{F}_p}(x-a) \quad \text{and} \quad (x^p - x)' = px^{p-1} - 1 \equiv -1 \not\equiv 0 \pmod{p},$$

so $x^p - x$ splits over $\mathbb{F}_p$ into *distinct* linear factors (it is square-free).

(F2) **Minimal-polynomial criterion.** A matrix A over a field K is diagonalizable over K iff its minimal polynomial $m_A(x)$ splits over K as a product of distinct linear factors (equivalently, m_A is square-free and has all roots in K).

Theorem 1. *Let $A \in \mathbb{F}_p^{n\times n}$. Then A is diagonalizable over $\mathbb{F}_p$ if and only if $A^p = A$.*

Proof. **(⇒)** If A is diagonalizable over $\mathbb{F}_p$, write $A = SDS^{-1}$ with $D = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$ and each $\lambda_i \in \mathbb{F}_p$. By (F1), $\lambda_i^p = \lambda_i$ for all i, hence

$$A^p = SD^pS^{-1} = S\operatorname{diag}(\lambda_1^p, \dots, \lambda_n^p)\,S^{-1} = S\operatorname{diag}(\lambda_1, \dots, \lambda_n)\,S^{-1} = A.$$

(⇐) If $A^p = A$, then A satisfies $f(x) = x^p - x$, i.e. $f(A) = 0$. Thus the minimal polynomial $m_A(x)$ divides $f(x)$. By (F1), f splits over $\mathbb{F}_p$ with no repeated roots, so m_A also splits over $\mathbb{F}_p$ and is square-free. By (F2), A is diagonalizable over $\mathbb{F}_p$. □

Corollary 1 (Invertible case)**.** *If A is invertible, then A is diagonalizable over $\mathbb{F}_p$ iff $A^{p-1} = I$.*

Proof. If A is diagonalizable, its eigenvalues lie in $\mathbb{F}_p^\times$, the cyclic group of order $p-1$, so $\lambda^{p-1} = 1$ and hence $A^{p-1} = I$. Conversely, if $A^{p-1} = I$, then $m_A(x) \mid x^{p-1} - 1$, which splits over $\mathbb{F}_p$ with distinct roots; by (F2), A is diagonalizable. □

1 Diagonalization Algorithm: Prelude

Prime and Precomputations

- **Field and constants:** prime p and a fixed generator g of $\mathbb{F}_p^\times$ (a *primitive root*).
- **Discrete logs:** a table $\texttt{seeds} : \mathbb{F}_p^\times \to \{1, \dots, p-1\}$ such that $g^{\texttt{seeds}[x]} = x$ for all $x \in \mathbb{F}_p^\times$.
- **d-th roots of unity:** for each $d \mid (p-1)$, define $\texttt{ones\_roots}[d] = \{\, g^{\,t(p-1)/d} \mid t = 0, \dots, d-1 \,\}$.
- **Factor schedule for $p-1$:** a list $\texttt{MOD\_decompose} = (d_1, \dots, d_m)$ of *prime* numbers (duplicates allowed) with $\prod_{i=1}^m d_i = p - 1$. Equivalently, if $p - 1 = \prod_{j=1}^r q_j^{e_j}$, then $\texttt{MOD\_decompose}$ is ascending ordering of the multiset containing e_j copies of each prime q_j (e.g., if $p = 101$, $\texttt{MOD\_decompose}$ is $(2, 2, 5, 5)$).
- **Field inverses:** a lookup table $\mathrm{inv}_p(a) = a^{-1} \bmod p$ for $a \in \{1, \dots, p-1\}$.

2 Diagonalization Algorithm: Implementation-Faithful Pseudocode

2.1 High-level idea

1. **Zero vs. nonzero split.** Compute $B := A^{p-1}$. Since for any eigenvalue λ of A we have $\lambda^{p-1} \in \{0, 1\}$, the spaces

   $$U_1 := \ker(B - I) \quad \text{and} \quad U_0 := \ker(B)$$

   capture, respectively, the direct sum of all nonzero eigenspaces and the 0-eigenspace. Form $S_0 = [\,\mathcal{B}(U_1)\ \mathcal{B}(U_0)\,]$, set $k := \dim U_1$, and obtain

   $$S_0^{-1} A S_0 = \begin{pmatrix} A_{\mathrm{nz}} & 0 \\ 0 & 0 \end{pmatrix}, \qquad A_{\mathrm{nz}} \in \mathbb{F}_p^{k \times k}.$$

2. **Power-map refinement on U_1.** Let $E \leftarrow p-1$ and keep a queue of current blocks $\mathcal{Q}$ (start with $\{A_{\mathrm{nz}}\}$). Maintain for each block $C \in \mathcal{Q}$ a *label* $\mu(C) \in \mathbb{F}_p^\times$ indicating a known eigenvalue of C^E. Initially $\mu(A_{\mathrm{nz}}) \leftarrow 1$. For each prime d in the schedule $\texttt{MOD\_decompose} = (d_1, \dots, d_m)$, do:

   - Replace $E \leftarrow E/d$.
   - For each block C (size m_C):
     - If $m_C = 1$: leave it as is.
     - If $m_C = 2$: diagonalize explicitly (closed form).
     - If $m_C \geq 3$: set $P := C^E$ and split P via the d preimages of $u(C)$ under $x \mapsto x^d$: choose a candidate set $\mathcal{R}_d(C) \subset \mathbb{F}_p^\times$ with $|\mathcal{R}_d(C)| = d$ such that $r^d = u(C)$ for every $r \in \mathcal{R}_d(C)$. For each candidate $\mathit{cand} \in \mathcal{R}_d(C)$, append the eigenspace $W = \ker(P - \mathit{cand} \cdot I)$. Concatenating bases of nonzero W gives a local similarity T_{loc} that block-diagonalizes C. The resulting diagonal blocks are re-enqueued, with labels set to their corresponding *cand*'s (which are eigenvalues of C^E).
   - Multiply an aggregate stitch matrix T_{acc} by each T_{loc} in place (this records the total similarity on U_1).

After exhausting `MOD_decompose`, every block in $\mathcal{Q}$ is 1×1 (true eigenspace).

3. **Assemble S and D.** Embed T_{acc} into the top-left $k \times k$ block of an $n \times n$ identity to get $E_{\text{emb}} = \operatorname{diag}(T_{\text{acc}}, I_{n-k})$, set $S := S_0 \, E_{\text{emb}}$ and write the k scalars from the final queue onto the first k diagonal entries of D (the remaining $n-k$ are 0 by construction).

2.2 Diagonalization Algorithm Pseudocode over $\mathbb{F}_p$

```
Input: $A \in \mathbb{F}_p^{n \times n}$; prime $p$, primitive generator $g \in \mathbb{F}_p^{\times}$; lookup tables seed$(\cdot)$, ones_roots$[\cdot]$, MOD_decompose (an ordered factorization of $p-1$); field inverse map $\mathrm{inv}_p(\cdot)$.
Output: $S \in \mathrm{GL}_n(\mathbb{F}_p)$, $D$ diagonal, with $S^{-1}AS = D$ (if $A$ is diagonalizable).

// Phase 1: split zero vs. nonzero spectrum
$B \leftarrow A^{p-1}$                                  ▷ $\lambda \mapsto \lambda^{p-1} \in \{0, 1\}$ over $\mathbb{F}_p$
$U_1 \leftarrow \mathrm{Null}(B - I)$;   $U_0 \leftarrow \mathrm{Null}(B)$     ▷ $U_1$: nonzero spectrum, $U_0$: 0-eigenspace
$S_0 \leftarrow [\, U_1 \;\; U_0 \,]$                  ▷ first $\dim U_1$ columns, then $\dim U_0$
$A_\star \leftarrow S_0^{-1} A S_0$                    ▷ $\left(\begin{smallmatrix} A_\star & * \\ 0 & 0 \end{smallmatrix}\right)$ with $A_\star$ invertible
$D \leftarrow 0_{n \times n}$
if $\dim U_1 = 0$ then
    return $(S_0, D)$

// Phase 2: refine the invertible block by power-splitting
$S_\star \leftarrow I_{n_1}$                           ▷ accumulated similarity on $A_\star$
$\mathcal{Q} \leftarrow$ queue with one item $A_\star$   ▷ worklist of current diagonal blocks
imgEig $\leftarrow$ list with one item 1               ▷ tracks $\mu = \lambda^\ell$ values seen at the current power level
$\ell \leftarrow p - 1$                                ▷ current exponent in $\mu = \lambda^\ell$; will be divided along MOD_decompose
for each $d$ in MOD_decompose do
    $\ell \leftarrow \ell / d$
    $T \leftarrow I_{n_1}$                             ▷ stage transform to be block-inserted into $S_\star$
    $m \leftarrow |\mathcal{Q}|$                       ▷ only process the blocks present at stage start
    offset $\leftarrow 0$                              ▷ where to place each block's local transform inside $T$
    for $t = 1$ to $m$ do
        Pop next block $X \in \mathbb{F}_p^{r \times r}$ from $\mathcal{Q}$, together with its tag $\mu \in$ imgEig (so far, $\mu = \lambda^{p-1}$ refined down to power $\ell d$)
        if $r = 1$ then
            // already a $1 \times 1$ block at this stage
            push $X$ back into $\mathcal{Q}$; append its scalar to imgEig; set $T_{\text{offset+!1, offset+!1}}! \leftarrow 1$; offset $\leftarrow$ offset $+ 1$; continue
        else if $r = 2$ then
            compute a closed-form diagonalization $X = S_2 D_2 S_2^{-1}$   ▷ explicit $2 \times 2$ formula over $\mathbb{F}_p$
            push $\big[D_2(1,1)\big]$, $\big[D_2(2,2)\big]$ into $\mathcal{Q}$; append the two scalars to imgEig
            write $S_2$ into the $2 \times 2$ block of $T$ at rows/cols offset!+!1 . . . offset!+!2; offset $\leftarrow$ offset $+ 2$; continue
        else
            // general $r \times r$ block: split by solving $(X^\ell - \gamma I)v = 0$
            $Y \leftarrow X^\ell$
            $\gamma_0 \leftarrow g^{\mathrm{seed}(\mu) \cdot \mathrm{inv}_p(d)}$      ▷ $\mu$ is eigenvalue of this block at power level $\ell$ from imgEig
            $\mathcal{B} \leftarrow [\,]$; $\Delta \leftarrow [\,]$
        for each $\zeta \in$ ones_roots$[d]$ do
            $\gamma \leftarrow \gamma_0 \cdot \zeta$                                        ▷ $\gamma^d = \mu$
            $W \leftarrow \mathrm{Null}\big(Y - \gamma I_r\big)$
            if $W \neq \{0\}$ then
                append columns of $W$ to $\mathcal{B}$; append $\dim W$ to $\Delta$
                append $\gamma$ to imgEig
        assert $\sum \Delta = r$                    ▷ if not same, not possible to diagonalize
        form $S_{\mathrm{loc}}$ from columns $\mathcal{B}$
        $X' \leftarrow S_{\mathrm{loc}}^{-1} X S_{\mathrm{loc}}$
        write $S_{\mathrm{loc}}$ into the block of $T$ at rows/cols offset+1 : offset+$r$; offset $\leftarrow$ offset + $r$
        chop $X'$ along sizes in $\Delta$ and push each block back into $\mathcal{Q}$
    $S_\star \leftarrow S_\star \cdot T$                 ▷ accumulate this stage's similarity on the invertible block
// Phase 3: read off eigenvalues and assemble the global similarity
// After the loop, every block in $\mathcal{Q}$ is $1 \times 1$; at the final level $\ell = 1$, its tag equals the true eigenvalue
let the scalars in imgEig (in queue order) be $\lambda_1, \ldots, \lambda_{n_1}$
for $i = 1$ to $n_1$ do
    $D(i,i) \leftarrow \lambda_i$
                                            ▷ zeros on indices $> n_1$ already set from Phase 1
$\widetilde{S} \leftarrow (\, S_\star \; 0 \; 0 \; I_{n_0} \,)$          ▷ pad refinement by identity on the 0-eigenspace
$S \leftarrow S_0 \cdot \widetilde{S}$
return $(S, D)$
```

2.3 Mathematical explanation of each phase

Phase 1 (split zero vs. nonzero spectrum; why the split is a *direct* sum, and why S_0 is invertible). Set $B := A^{p-1}$. For any eigenpair (λ, v) of A,

$$Bv \;=\; A^{p-1}v \;=\; \lambda^{p-1}v \;=\; \begin{cases} 0 \cdot v, & \lambda = 0, \\ 1 \cdot v, & \lambda \in \mathbb{F}_p^{\times}. \end{cases}$$

Thus the spectrum of B is contained in $\{0,1\}$, so its minimal polynomial divides $x(x-1)$. Since x and $x-1$ are coprime, the primary decomposition theorem yields

$$\mathbb{F}_p^n \;=\; \ker(B) \;\oplus\; \ker(B - I).$$

Write $U_0 := \ker(B)$, $U_1 := \ker(B - I)$, and $k := \dim U_1$. Choose bases $\mathcal{B}(U_1)$, $\mathcal{B}(U_0)$ and set

$$S_0 \;=\; \big[\,\mathcal{B}(U_1) \;\; \mathcal{B}(U_0)\,\big] \in \mathrm{GL}_n(\mathbb{F}_p).$$

Invertibility of S_0 follows because the sum is direct and spans all of $\mathbb{F}_p^n$. In this basis,

$$S_0^{-1} A S_0 \;=\; \begin{pmatrix} A_\star & * \\ 0 & A|_{U_0} \end{pmatrix}, \qquad A_\star := A|_{U_1} \in \mathbb{F}_p^{k \times k}.$$

If A is diagonalizable over $\mathbb{F}_p$ (equivalently, its minimal polynomial divides the square-free polynomial $x(x^{p-1} - 1)$), then $A|_{U_0} = 0$ and the bottom-right block is already diagonal (all zeros). Hence the problem reduces to diagonalizing the invertible block $A_\star$ on U_1.

Phase 2 (refine the nonzero block). Let $\mathcal{Q}$ be a queue of current blocks, initialized as $\mathcal{Q} = \{A_\star\}$. Maintain an accumulated similarity $S_\star := I_k$. Because $\mathbb{F}_p^\times$ is cyclic of order $p-1$, there is a primitive generator g. At any stage we fix an exponent ℓ (starting at $\ell = p-1$) and attach to each block $X \in \mathcal{Q}$ a tag $\mu = \lambda^\ell$ recording the image of its eigenvalues under the power map. For the next factor d in `MOD_decompose`, set $\ell \leftarrow \ell/d$ and split each block X by solving

$$\big(X^\ell - \gamma I\big)v = 0$$

for all d distinct γ with $\gamma^d = \mu$. Concretely, choose one d-th root

$$\gamma_0 \;=\; g^{\,\mathrm{seed}(\mu)\cdot \mathrm{inv}_p(d)},$$

and enumerate the full coset $\gamma = \gamma_0\zeta$ with $\zeta \in \text{ones\_roots}[d]$. For each such γ, the kernel $\ker\big(X^\ell - \gamma I\big)$ is an X-invariant subspace; collecting the nonzero kernels gives a direct sum that exhausts the block. Writing the columns of a local basis into S_{loc} yields

$$S_{\mathrm{loc}}^{-1} X S_{\mathrm{loc}} \;=\; \mathrm{blockdiag}(X_1, \ldots, X_m),$$

with strictly smaller blocks X_i. We "stitch" all local transforms into a stage matrix T (block-inserted on the diagonal), update $S_\star \leftarrow S_\star T$, and replace X in $\mathcal{Q}$ by its diagonal blocks X_i. A 2×2 block is handled once by a closed-form diagonalization; 1×1 blocks are already eigenblocks. Iterating over all $d \in$ `MOD_decompose` forces ℓ down to 1 and yields only 1×1 blocks.

Final assembly (how blocks flow through $\mathcal{Q}$ and how T, $S_\star$ build the final S and D). Each outer round takes a snapshot of $\mathcal{Q}$, applies the local splitters S_{loc} block-diagonally (forming the stage matrix T), updates $S_\star \leftarrow S_\star T$, and replaces the processed block by its children. When the schedule ends ($\ell = 1$), every block in $\mathcal{Q}$ is 1×1; the final tags equal the true eigenvalues $\lambda_1, \ldots, \lambda_k$. Set the top-left $k \times k$ diagonal of D to these λ_i (and the remaining diagonal entries to 0), and define

$$S \;=\; S_0 \begin{pmatrix} S_\star & 0 \\ 0 & I_{n-k} \end{pmatrix}.$$

Under the diagonalizability hypothesis, this yields $S^{-1}AS = D$.

3 Diagonalization Algorithm: Time Complexity

Let $k = \dim\ker(A^{p-1} - I)$ be the size of the nonzero block and let the schedule be $d_1, \ldots, d_L$ with $\prod_{r=1}^{L} d_r = p-1$. Write the block sizes at the start of round r as $m_1^{(r)}, \ldots, m_{q_r}^{(r)}$ (so $\sum_j m_j^{(r)} = k$). We assume dense arithmetic where a nullspace solve or an inverse on an $m \times m$ matrix costs $O(m^3)$.

Phase 1 (one–time costs).

- Two nullspace computations on $n \times n$: $\ker(A^{p-1} - I)$ and $\ker(A^{p-1}) \Rightarrow 2\,O(n^3)$.
- One inverse of the assembled $S_0 \in \mathrm{GL}_n \Rightarrow O(n^3)$.

After this reduction, work proceeds on the $k \times k$ invertible block; subsequent costs depend on the $m_j^{(r)}$.

Per round r (only counting nullspaces & inverses). For each current block $X \in \mathbb{F}_p^{m \times m}$ at the start of round r:

- **Nullspaces:** exactly d_r candidate tests $\ker\big(X^{\ell_r} - \gamma I\big) \Rightarrow d_r$ nullspace solves on $m \times m$ matrices $\rightsquigarrow\ d_r\, O(m^3)$.
- **Inverse:** one inverse of the local splitter $S_{\text{loc}} \in \mathbb{F}_p^{m\times m} \Rightarrow O(m^3)$.

Summing over blocks and using $\sum_j (m_j^{(r)})^3 \le k^3$,

$$\sum_{j=1}^{q_r} \Big(d_r\, O\big((m_j^{(r)})^3\big) + O\big((m_j^{(r)})^3\big) \Big) \ \le\ O\big((d_r+1)\, k^3\big).$$

(Blocks of size 1×1 incur zero cost; 2×2 use a closed form with $O(1)$ overhead—ignored here.)

Sequential total. Across all rounds,

$$T_{\text{seq}}(n,p) \ =\ O(n^3)\ +\ \sum_{r=1}^{L} O\big((d_r+1)\, k^3\big)\ =\ O\Big(n^3 + k^3\Big(L + \textstyle\sum_{r=1}^{L} d_r\Big)\Big).$$

For a fixed prime p (so L and $\sum_r d_r$ depend only on p), this simplifies to

$$T_{\text{seq}}(n,p) \ =\ O(n^3) \quad \text{since } k \le n.$$

This matches the cubic profile of dense inversion. The actual runtime results are summarized below.

Parallel (GPU) time

Within a round, two independent axes: (i) across blocks $j = 1, \ldots, q_r$; (ii) within each block, across the d_r candidate nullspace solves. Let $C_{\text{blk}}(r) \le q_r$ be effective block concurrency and $C_{\text{cand}}(r) \le d_r$ candidate concurrency. Ignoring multiplies and counting only solves/inverses, an idealized round time is

$$T_r \ =\ O\bigg(\frac{k^3\, d_r}{C_{\text{blk}}(r)\, C_{\text{cand}}(r)}\bigg)\ +\ O\bigg(\frac{k^3}{C_{\text{blk}}(r)}\bigg).$$

As rounds progress, blocks shrink (cheaper solves/inverses) and their count grows (larger $C_{\text{blk}}(r)$), so T_r decreases. Summing over rounds,

$$T_{\text{par}}(n,p) \ \lesssim\ O\Big(k^3 \sum_{r=1}^{L} \frac{d_r}{C_{\text{blk}}(r)\, C_{\text{cand}}(r)}\ +\ k^3 \sum_{r=1}^{L} \frac{1}{C_{\text{blk}}(r)}\Big),$$

which exhibits the same n^3 scaling, up to constants driven by p and achievable concurrency.

Table 1: Runtime comparison across moduli: `matrix_inverse` vs. `matrix_diagonalize_henry` on $\mathbb{F}_p$. Each (p, N) entry is averaged over 100 trials. Speedup is Avg inv / Avg diag. Random matrices used for the experiment follow uniform distribution on $\mathbb{F}_p$

p	N	Avg inv (s)	Avg diag (s)	Speedup
65537	10	0.000020	0.000925	0.022101
65537	20	0.000124	0.006209	0.020043
65537	30	0.000407	0.020664	0.019689
65537	50	0.001833	0.092663	0.019778
65537	100	0.013996	0.729230	0.019192
131071	10	0.000017	0.001049	0.016610
131071	20	0.000127	0.007815	0.016226
131071	30	0.000404	0.025404	0.015907
131071	50	0.001759	0.116970	0.015041
131071	100	0.013914	0.929978	0.014961
524287	10	0.000019	0.001165	0.016279
524287	20	0.000122	0.008607	0.014227
524287	30	0.000385	0.028076	0.013720
524287	50	0.001753	0.129695	0.013515
524287	100	0.013923	1.037240	0.013423
653659	10	0.000015	0.054949	0.000279
653659	20	0.000117	0.246325	0.000476
653659	30	0.000380	0.569895	0.000667
653659	50	0.001747	1.779385	0.000981
653659	100	0.013835	10.521617	0.001314
100000007	10	0.000030	0.006967	0.004253
100000007	20	0.000123	0.015248	0.008081
100000007	30	0.000391	0.044541	0.008767
100000007	50	0.001752	0.188656	0.009284
100000007	100	0.013862	1.440774	0.009621

Note. For fixed p, the Speedup stays roughly constant (and sometimes increases) as N grows, which is consistent with both inversion and diagonalization exhibiting $O(n^3)$ scaling.

Table 2: Factorizations of $p-1$ for selected primes p used in the benchmarks.

p	Factorization of $p-1$	Notes
65537	2^{16}	Fermat prime; $p-1$ is a power of two.
131071	$2 \cdot 3 \cdot 5 \cdot 17 \cdot 257$	$p = 2^{17} - 1$ (Mersenne prime).
524287	$2 \cdot 3^3 \cdot 7 \cdot 19 \cdot 73$	$p = 2^{19} - 1$; "well distributed" factors.
653659	$2 \cdot 3 \cdot 108{,}943$	108,943 is prime.
100000007	$2 \cdot 491 \cdot 101{,}833$	Poorly distributed.

Note. For the Fermat prime 65537, the factorization $p - 1 = 2^{16}$ contains only small factors. Consequently, each decomposition round can split the matrix only by two or not at all, causing the block size to shrink slowly, making Speedup decreases with N. In contrast, primes such as $p = 653659, 100000007$ have large number as factor, allowing deeper factor-driven splitting. These cases benefit more strongly when N is large. Nevertheless, well distributed prime's diagonalization takes less time.