Lanczos Method for Eigensystems

Aiichiro Nakano

Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Chemical Engineering & Materials Science
Department of Biological Sciences
University of Southern California

Email: anakano@usc.edu

Another winning *O*(*N*) approach (in addition to divide-&-conquer tree) by subspace projection

B. N. Parlett
The Symmetric Eigenvalue Problem
(Prentice-Hall, '80) Secs. 11-13

History's Top 10 Algorithms Again

In putting together this issue of *Computing in Science & Engineering*, we knew three things: it would be difficult to list just 10 algorithms; it would be fun to assemble the authors and read their papers; and, whatever we came up with in the end, it would be controversial. We tried to assemble the 10 algorithms with the greatest influence on the development and practice of science and engineering in the 20th century. Following is our list (here, the list is in chronological order; however, the articles appear in no particular order):

PHYS 516

CSCI 653

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

IEEE Comput. Sci. Eng. **2(1)**, 22 ('00)

Rayleigh Quotient

Theorem

Let A be an $n \times n$ real symmetric matrix, $\lambda_1[A] \le \ldots \le \lambda_n[A]$ its eigenvalues in ascending order, $x \in R^n$, & the Rayleigh quotient

$$\rho(\mathbf{x};\mathbf{A}) = \frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{x}} \quad \text{then} \quad \begin{cases} \lambda_1[\mathbf{A}] = \min\limits_{\mathbf{x}\in\Re^n} \rho(\mathbf{x};\mathbf{A}) \\ \lambda_n[\mathbf{A}] = \max\limits_{\mathbf{x}\in\Re^n} \rho(\mathbf{x};\mathbf{A}) \end{cases}$$

Proof

Let $q^{(k)}$ be the k-th orthonormalized eigenvector of A, $\mathbf{A}\mathbf{q}_k = \lambda_k \mathbf{q}_k$ **, & orthogonal transformation matrix,** $\mathbf{Q} = [\mathbf{q}_1\mathbf{q}_2 \cdots \mathbf{q}_n]$ **, then**

$$\mathbf{Q}^T\mathbf{A}\mathbf{Q} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

Let $x = Qz$ (note $Q^TQ = I$), then

$$\rho(\mathbf{x};\mathbf{A}) = \frac{\mathbf{z}^T\mathbf{Q}^T\mathbf{A}\mathbf{Q}\mathbf{z}}{\mathbf{z}^T\mathbf{Q}^T\mathbf{Q}\mathbf{z}} = \frac{z_1^2\lambda_1 + \cdots \; z_n^2\lambda_n}{z_1^2 + \cdots \; z_n^2}$$

which is a weighted average of $\lambda_1, \ldots, \lambda_n$, & the minimum is when $z^T = (1,0,\ldots,0) = e_1$ & $x = Qe_1 = q_1$.

Rayleigh-Ritz Procedure

Theorem

Let $\{q_1,\ldots,q_m\}$ be an orthonormal set that spans R^m $(m < n) \subset R^n$, so that any vector $x \in R^m$ is expressed as a linear combination of $q_1,\ldots,q_m$:

$$\mathbf{x} = z_1\mathbf{q}_1 + \cdots + z_m\mathbf{q}_m \qquad \text{or} \qquad \begin{array}{c c} & 1 \\ n & \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \end{array} = \begin{array}{c c} & m \\ n & \begin{bmatrix} \mathbf{q}_1 & \cdots & \mathbf{q}_m \end{bmatrix} \end{array} \begin{array}{c c} 1 & \\ \begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix} & m \end{array} = \mathbf{Qz}$$

then the best approximations for $\lambda_1[A]$ & $\lambda_n[A]$ are obtained by diagonalizing

$$\underset{m\times m}{\mathbf{H}} = \underset{m\times n}{\mathbf{Q}^T}\ \underset{n\times n}{\mathbf{A}}\ \underset{n\times m}{\mathbf{Q}}$$

as $\lambda_1[H]$ & $\lambda_m[H]$.

Proof

Note $\left(\mathbf{Q}^T\mathbf{Q}\right)_{ij} = \sum_{k=1}^{n} Q_{ki}Q_{kj} = \mathbf{q}_i \bullet \mathbf{q}_j = \delta_{ij} \quad 1 \le i, j \le m$

then

$$\rho(\mathbf{x};\mathbf{A}) = \frac{\mathbf{z}^T\mathbf{Q}^T\mathbf{AQz}}{\mathbf{z}^T\mathbf{Q}^T\mathbf{Qz}} = \frac{\mathbf{z}^T\mathbf{Hz}}{\mathbf{z}^T\mathbf{z}} = \frac{z_1^2\lambda_1(H) + \ldots + z_m^2\lambda_m(H)}{z_1^2 + \ldots + z_m^2}$$

the minimum of which is $\lambda_1[H]$.

Orthogonalization by QR Decomposition

- **Gram-Schmidt orthonormalization: The orthonormal set Q required for the Rayleigh-Ritz procedure is obtained starting from an arbitrary set of *m* vectors, S = [$s_1 \ldots s_m$] ($s_j \in R^n$) as:**

$$
\begin{aligned}
&\mathbf{q}_1 = \mathbf{s}_1 / |\mathbf{s}_1| \\
&\text{for} \quad i = 2 \quad \text{to} \quad m \\
&\qquad \mathbf{q}'_i = \mathbf{s}_i - \sum_{j=1}^{i-1} \mathbf{q}_j \left(\mathbf{q}_j \bullet \mathbf{s}_i \right) \\
&\qquad \mathbf{q}_i = \mathbf{q}'_i / |\mathbf{q}'_i| \\
&\text{endfor}
\end{aligned}
$$

Projection!

- **The Gram-Schmidt amounts to QR decomposition, S = QR, where R is an *m*×*m* right-triangle matrix:**

$$
\begin{array}{c}
 \\ n
\end{array}
\overset{m}{\begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \mathbf{s}_3 & \mathbf{s}_4 \end{bmatrix}}
= n \overset{m}{\begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \mathbf{q}_3 & \mathbf{q}_4 \end{bmatrix}}
\overset{m}{\begin{bmatrix}
|\mathbf{q}'_1| & \mathbf{q}_1 \bullet \mathbf{s}_2 & \mathbf{q}_1 \bullet \mathbf{s}_3 & \mathbf{q}_1 \bullet \mathbf{s}_4 \\
0 & |\mathbf{q}'_2| & \mathbf{q}_2 \bullet \mathbf{s}_3 & \mathbf{q}_2 \bullet \mathbf{s}_4 \\
0 & 0 & |\mathbf{q}'_3| & \mathbf{q}_3 \bullet \mathbf{s}_4 \\
0 & 0 & 0 & |\mathbf{q}'_4|
\end{bmatrix}} \; m
$$

$$
\because \quad \mathbf{s}_i = |\mathbf{q}'_i| \mathbf{q}_i + \sum_{j=1}^{i-1} \mathbf{q}_j \left(\mathbf{q}_j \bullet \mathbf{s}_i \right)
$$

Rayleigh-Ritz Algorithm

1. Start from $\mathbf{S} = [\mathbf{s}_1 \ldots \mathbf{s}_m]$ ($\mathbf{s}_j \in \mathbf{R}^n$) & do Gram-Schmidt orthonormalization, $\mathbf{S} = \mathbf{QR}$, to obtain an orthonormal set $\mathbf{Q} = [\mathbf{q}_1 \ldots \mathbf{q}_m]$

2. Form $\mathbf{H} = \mathbf{Q}^{\mathrm{T}}\mathbf{AQ}$

3. Diagonalize $\mathbf{H}$ to get $\lambda_1[\mathbf{H}], \ldots, \lambda_m[\mathbf{H}]$: $\quad \mathbf{Hg}_k = \lambda_k[\mathbf{H}]\mathbf{g}_k \quad (k = 1, \ldots, m)$

4. The approximations of $\lambda_1[\mathbf{A}]$ & $\lambda_n[\mathbf{A}]$ are given by $\lambda_1[\mathbf{H}]$ & $\lambda_m[\mathbf{H}]$ with the corresponding eigenvectors, $\mathbf{y}_k = \mathbf{Qg}_k$ ($k = 1$ & m).

$$\underbrace{\mathbf{Q}^T\mathbf{AQ}}_{\mathbf{H}}\,\mathbf{g}_k = \lambda_k[\mathbf{H}]\mathbf{g}_k$$

$$\downarrow \mathbf{Q}\times$$

$$\mathbf{A}\,\underbrace{\mathbf{Qg}_k}_{\mathbf{y}_k} = \lambda_k[\mathbf{H}]\,\underbrace{\mathbf{Qg}_k}_{\mathbf{y}_k}$$

Krylov Subspace

- **Krylov subspace S_m is spanned by a Krylov matrix, $K^m(f) = [f\ Af \ldots A^{m-1}f]$ ($f \in R^n$)**

Theorem

Let Q_m be the orthonormal basis obtained by QR factorization, $K_m(f) = Q_m R$, then $T_m = Q_m^T A Q_m$ is a tridiagonal matrix

Proof

For $i > j+1$, $q_i^T(Aq_j) = 0$, since $Aq_j \subset S_{j+1}$ by construction & $q_i \perp S_{j+1}$ by Gram-Schmidt orthonormalization for $i > j+1$. By the symmetry of A, $q_i^T(Aq_j) = q_j^T(A^Tq_i) = q_j^T(Aq_i) = 0$ for $j > i+1$ or $i < j-1$.

$$T_m = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{m-2} & \alpha_{m-1} & \beta_{m-1} \\ & & & \beta_{m-1} & \alpha_m \end{bmatrix} \qquad \begin{cases} \alpha_j = \mathbf{q}_j^T \mathbf{A} \mathbf{q}_j & j = 1, \ldots, m \\ \beta_j = \mathbf{q}_{j+1}^T \mathbf{A} \mathbf{q}_j & j = 1, \ldots, m-1 \end{cases}$$

- **Tridiagonal matrix can be diagonalized in $O(N)$ time**

Alexei Krylov with daughter Anna, later Anna Kapitsa, wife of Pyotr Kapitsa (1904)

Recursion Formula

- **Due to the tridiagonality, A$\mathbf{q}_i$ is a linear combination of $\mathbf{q}_{i-1}$, $\mathbf{q}_i$ & $\mathbf{q}_{i+1}$:**

$$\mathbf{A}\mathbf{q}_i = \beta_{i-1}\mathbf{q}_{i-1} + \alpha_i \mathbf{q}_i + \beta_i \mathbf{q}_{i+1} \quad (2 \le i \le m-1)$$

If we define $\mathbf{q}_0 = 0$, the above equation is valid for $i = 1$ as well. Let $\mathbf{r}_i \equiv \beta_i \mathbf{q}_{i+1}$ ($\mathbf{r}_i$ is a component of A$\mathbf{q}_i$ orthogonal to $\mathbf{q}_j$ for $j \le i$), then

$$\mathbf{r}_i = \mathbf{A}\mathbf{q}_i - \beta_{i-1}\mathbf{q}_{i-1} - \alpha_i \mathbf{q}_i \quad (1 \le i \le m-1)$$

- **Lanczos algorithm:**

$$
\begin{aligned}
&\text{Given} \quad \mathbf{r}_0, \beta_0 = \|\mathbf{r}_0\| \quad (\mathbf{q}_0 = 0) \\
&\text{for} \quad i = 1, \ldots, m \\
&\qquad \mathbf{q}_i \leftarrow \mathbf{r}_{i-1} / \beta_{i-1} \\
&\qquad \mathbf{r}_i \leftarrow \mathbf{A}\mathbf{q}_i - \beta_{i-1}\mathbf{q}_{i-1} \\
&\qquad \alpha_i \leftarrow \mathbf{q}_i^T \mathbf{r}_i \qquad \because \mathbf{q}_i^T(\mathbf{A}\mathbf{q}_i - \beta_{i-1}\mathbf{q}_{i-1}) = \mathbf{q}_i^T \mathbf{A}\mathbf{q}_i = \alpha_i \quad \text{(orthogonality)} \\
&\qquad \mathbf{r}_i \leftarrow \mathbf{r}_i - \alpha_i \mathbf{q}_i \\
&\qquad \beta_i = \|\mathbf{r}_i\| \quad (\text{only} \quad \text{when} \quad i \le m-1) \\
&\text{endfor}
\end{aligned}
$$

Keep increasing m until $\lambda_1[\mathrm{T}_m]$ converges

An Application of Rayleigh-Ritz/Lanczos

- **Search for transition states (with a negative eigenvalue of the Hessian matrix, $\partial^2E/\partial r_i \partial r_j$, by following the eigenvector with the smallest eigenvalue**
  - **Rayleigh-Ritz:** Kumeda, Wales & Munro, *Chem. Phys. Lett*. **341**, 185 ('01)
  - **Lanczos:** Mousseau *et al*., *J. Mol. Graph. Model.* **19**, 78 ('01)

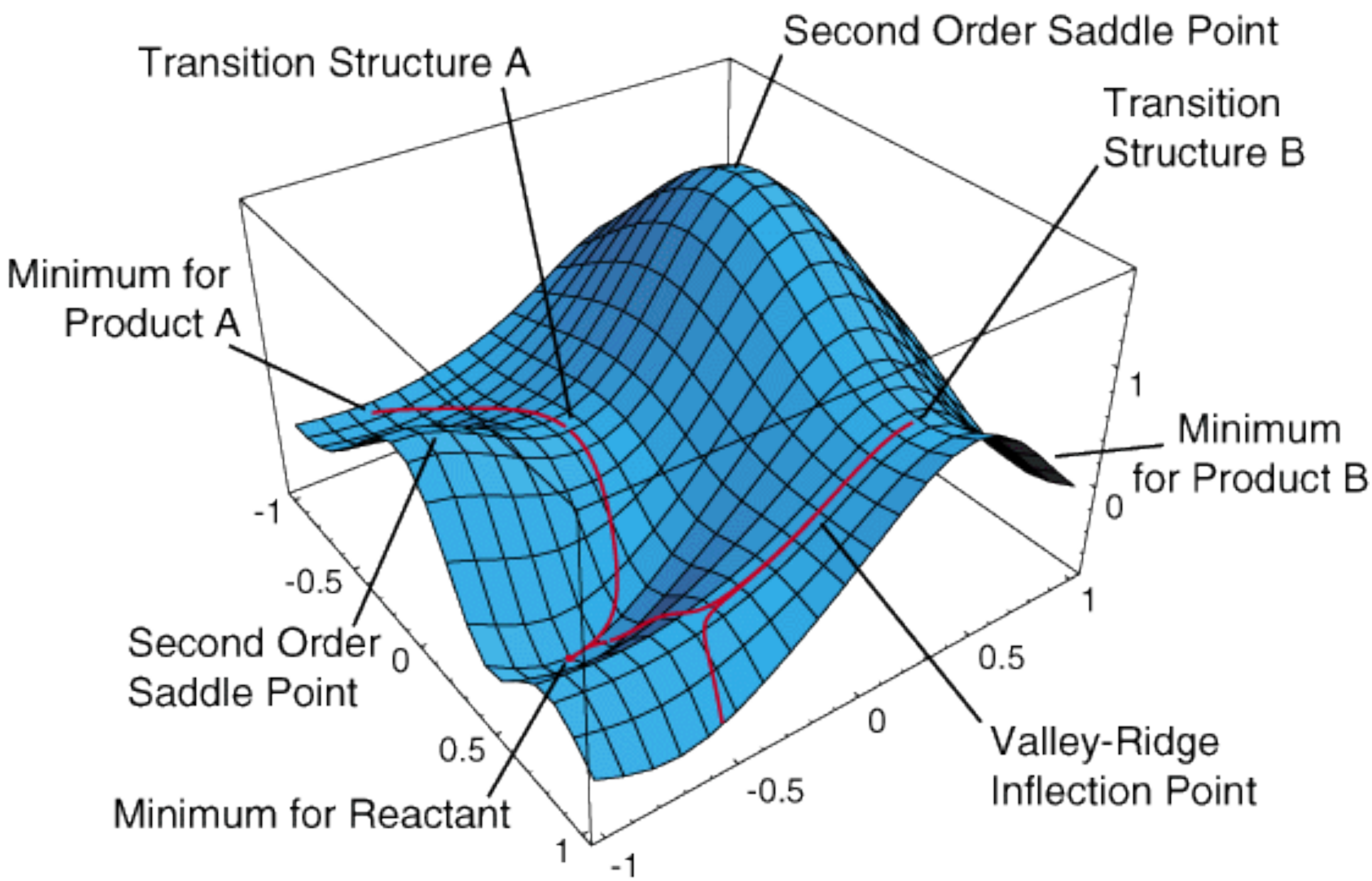


Figure from Prof. H. B. Schlegel; http://chem.wayne.edu/schlegel

Lanczos Algorithm for Hessian Calculation

A. Nakano / Computer Physics Communications 176 (2007) 292–299

Algorithm Lanczos

Input:

$\mathbf{R} \in \mathbb{R}^{3N}$: a state

logical *initialize*: TRUE for the first call in each event generation; FALSE otherwise

Output:

λ_1: the minimum eigenvalue of the Hessian matrix, $\mathbf{H}(\mathbf{R}) = \partial^2 V/\partial \mathbf{R}^2$

$\mathbf{V}^1 \in \mathbb{R}^{3N}$: the Hessian eigenvector corresponding to λ_1

Steps:

if *initialize*

randomize $\Delta \in \mathbb{R}^{3N}$, such that it contains no translational motion

$s \leftarrow 0$

$\beta^s \leftarrow \|\Delta\|$

$\mathbf{Q}^s \ (\in \mathbb{R}^{3N}) \leftarrow 0$

do

$s \leftarrow s + 1$

$\mathbf{Q}^s \leftarrow \Delta/\beta^{s-1}$

$c_{\text{fd}} \leftarrow \max_{i\alpha}\{|q^s_{i\alpha}| \mid i = 1, \ldots, N; \alpha = x, y, z\}/\delta_{\text{fd}}$

$\Delta \leftarrow c_{\text{fd}}[-\mathbf{F}(\mathbf{R} + \mathbf{Q}^s/c_{\text{fd}}) + \mathbf{F}(\mathbf{R})] - \beta^{s-1}\mathbf{Q}^{s-1}$

$\alpha^s \leftarrow \mathbf{Q}^{s\mathrm{T}}\Delta$

$\Delta \leftarrow \Delta - \alpha^s \mathbf{Q}^s$

$\beta^s \leftarrow \|\Delta\|$

diagonalize $\mathbf{T}_s = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{s-2} & \alpha_{s-1} & \beta_{s-1} \\ & & & \beta_{s-1} & \alpha_s \end{bmatrix}$, so that $\tilde{\mathbf{Q}}_s^{\mathrm{T}}\mathbf{T}_s\tilde{\mathbf{Q}}_s = \mathrm{diag}(\tilde{\lambda}_1^s, \ldots, \tilde{\lambda}_s^s)^*$

while $|(\tilde{\lambda}_1^s - \tilde{\lambda}_1^{s-1})/\tilde{\lambda}_1^{s-1}| > \Delta_{\text{eigen}}$

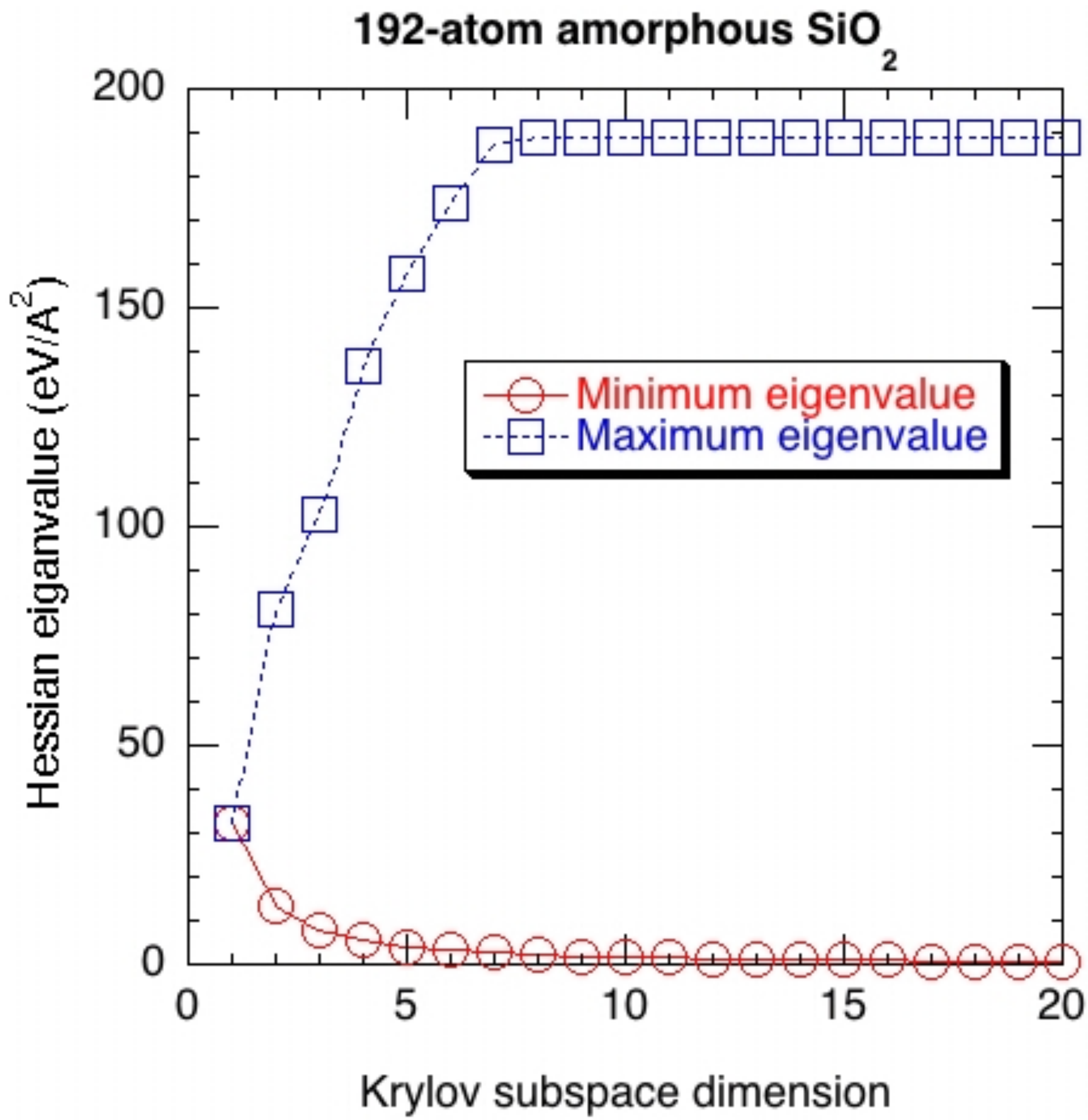
$\lambda_1 \leftarrow \tilde{\lambda}_1^s$

$\mathbf{V}^1 \leftarrow \sum_{k=1}^{s} \mathbf{Q}^k \tilde{q}_k^1$

$\mathbf{V}^1 \leftarrow \mathbf{V}^1/\|\mathbf{V}^1\|$

* $\mathrm{diag}(\tilde{\lambda}_1^s, \ldots, \tilde{\lambda}_s^s)$ is an s by s diagonal matrix, with its diagonal elements given by $\tilde{\lambda}_1^s, \ldots, \tilde{\lambda}_s^s$. $\tilde{\mathbf{Q}}^s = [\tilde{\mathbf{q}}^1, \ldots, \tilde{\mathbf{q}}^s]$ is an s by s orthogonal matrix, with $\tilde{\mathbf{q}}^m \in \mathbb{R}^s$ is the mth eigenvector of $\mathbf{T}_s$.

Sample Run of Lanczos Program



Electronic Energy Bands of GaAs

- **8-band k•p model**

$$H_k=\begin{pmatrix} A & 0 & V^* & 0 & \sqrt{3}V & -\sqrt{2}U & -U & \sqrt{2}V^* \\ 0 & A & -\sqrt{2}U & -\sqrt{3}V^* & 0 & -V & \sqrt{2}V & U \\ V & -\sqrt{2}U & -P+Q & -S^* & R & 0 & \sqrt{\frac{3}{2}}\,S & -\sqrt{2}Q \\ 0 & -\sqrt{3}V & -S & -P-Q & 0 & R & -\sqrt{2}R & \frac{1}{\sqrt{2}}S \\ \sqrt{3}V^* & 0 & R^* & 0 & -P-Q & S^* & \frac{1}{\sqrt{2}}S^* & \sqrt{2}R^* \\ -\sqrt{2}U & -V^* & 0 & R^* & S & -P+Q & \sqrt{2}Q & \sqrt{\frac{3}{2}}\,S^* \\ -U & \sqrt{2}V^* & \sqrt{\frac{3}{2}}\,S^* & -\sqrt{2}R^* & \frac{1}{\sqrt{2}}S & \sqrt{2}Q & -P-\Delta & 0 \\ \sqrt{2}V & U & -\sqrt{2}Q & \frac{1}{\sqrt{2}}S^* & \sqrt{2}R & \sqrt{\frac{3}{2}}\,S & 0 & -P-\Delta \end{pmatrix}$$

$$A=E_c-\frac{\hbar^2}{2m_0}(\partial_x^2+\partial_y^2+\partial_z^2),$$

$$P=-E_v-\gamma_1\frac{\hbar^2}{2m_0}(\partial_x^2+\partial_y^2+\partial_z^2),$$

$$Q=-\gamma_2\frac{\hbar^2}{2m_0}(\partial_x^2+\partial_y^2-2\partial_z^2),$$

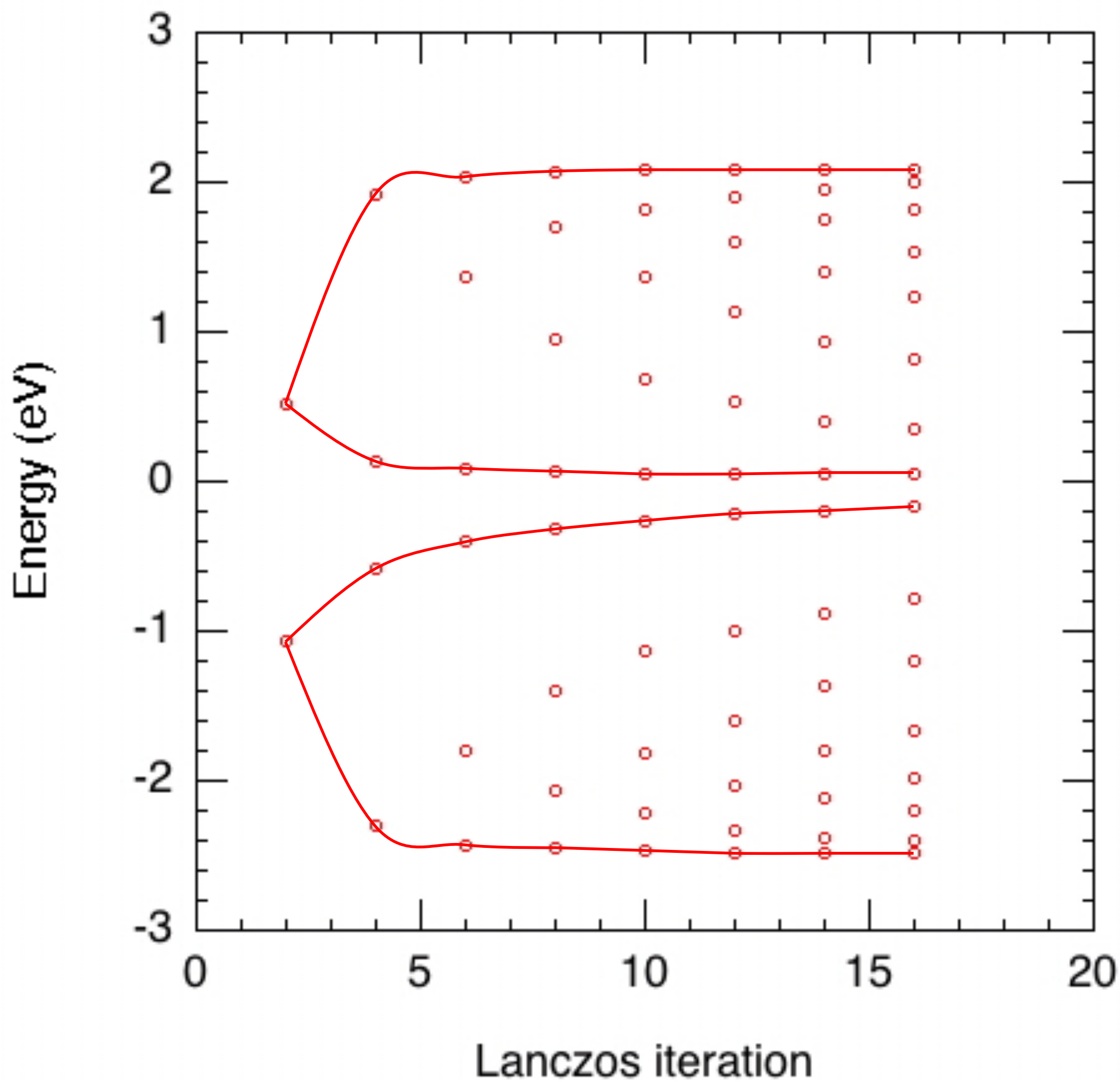
$$R=\sqrt{3}\,\frac{\hbar^2}{2m_0}[\gamma_2(\partial_x^2-\partial_y^2)-2i\gamma_3\partial_x\partial_y],$$

$$S=-\sqrt{3}\gamma_3\frac{\hbar^2}{m_0}\partial_z(\partial_x-i\partial_y),$$

$$U=\frac{-i}{\sqrt{3}}P_0\partial_z\,,$$

$$V=\frac{-i}{\sqrt{6}}P_0(\partial_x-i\partial_y).$$

C. Pryor, *Phys. Rev. B* **57**, 7190 ('98)



Lanczos Program in Fortran

```
do s = 1,NWF
  q(:,:,:,s) = v/bet(s-1)
  call hamiltonian_op(q(:,:,:,s),hv)  ! Operates Hamiltonian H on Q(S)
  v = hv-bet(s-1)*q(:,:,:,s-1)
  alp(s) = inner_product(q(:,:,:,s),v)
  v = v-alp(s)*q(:,:,:,s)
  bet(s) = sqrt(inner_product(v,v))
  call tridiag(eval,s)  ! Diagonalize the S by S tridiagonal matrix
end do  ! Lanczos iteration S
```

Given $\mathbf{r}_0, \beta_0 = \|\mathbf{r}_0\| \quad (\mathbf{q}_0 = 0)$

for $i = 1, \ldots, m$

$\quad \mathbf{q}_i \leftarrow \mathbf{r}_{i-1} / \beta_{i-1}$

$\quad \mathbf{r}_i \leftarrow \mathbf{A}\mathbf{q}_i - \beta_{i-1}\mathbf{q}_{i-1}$

$\quad \alpha_i \leftarrow \mathbf{q}_i^T \mathbf{r}_i$

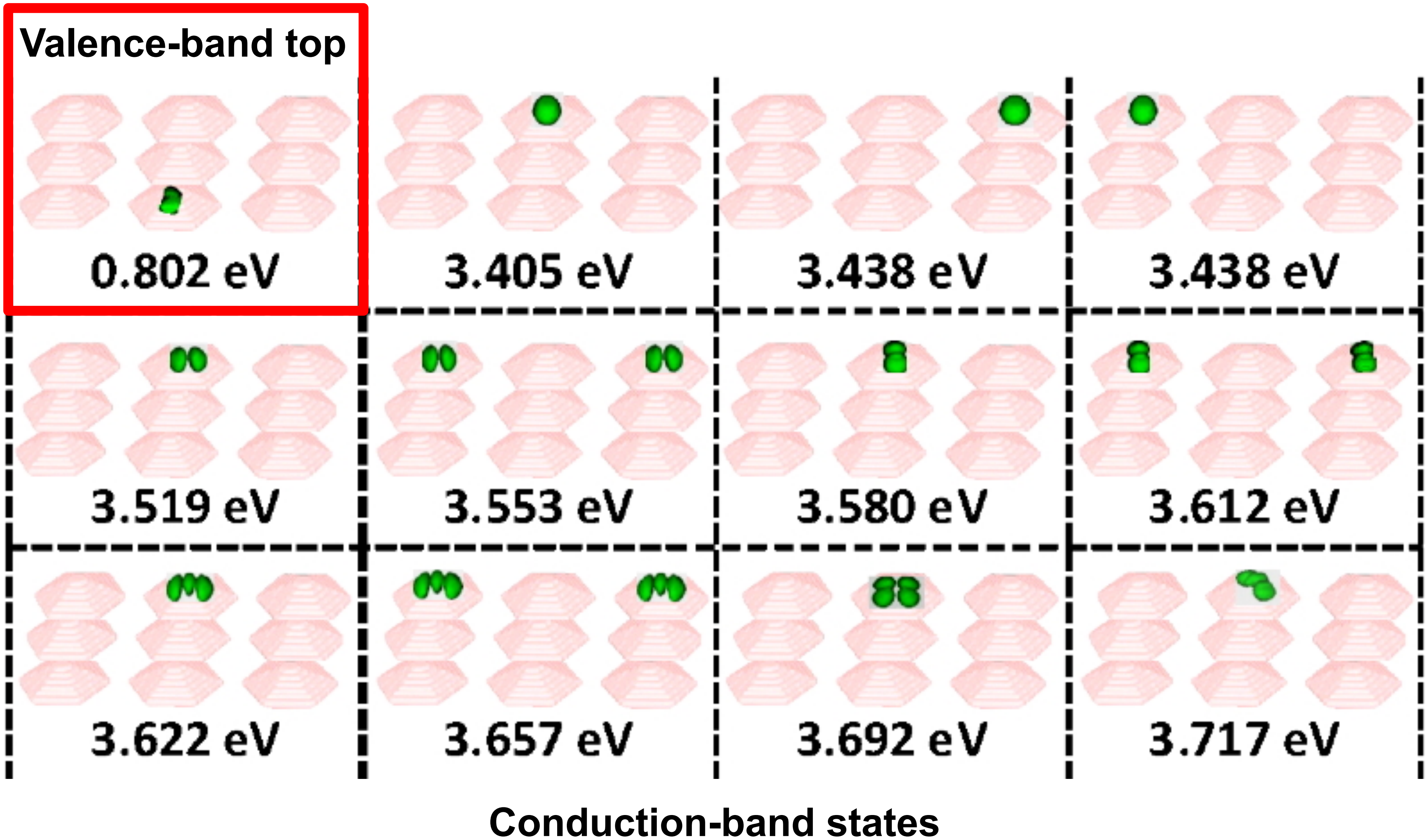
$\quad \mathbf{r}_i \leftarrow \mathbf{r}_i - \alpha_i \mathbf{q}_i$

$\quad \beta_i = \|\mathbf{r}_i\| \quad (\text{only} \quad \text{when} \quad i \le m-1)$

endfor

Band-edge Wave Functions

- Band-edge states in an array of GaN quantum dots in AlN matrix



S. Sburlan, Ph.D. dissertation, USC ('13)