



Block-Diagonal Decomposition of Matrices based on $*$ -Algebra for Structural Optimization with Symmetry Property

Yoshihiro Kanno[†] Kazuo Murota[†]
Masakazu Kojima[‡] Sadayoshi Kojima[‡]

[†]University of Tokyo (Japan)
[‡]Tokyo Institute of Technology (Japan)



June 16, 2008

simultaneous block-diagonal decomposition

- given: A_1, A_2, \dots, A_m — symmetric $n \times n$
- find: $P \in \mathbb{R}^{n \times n}$ — orthogonal

$$A_1 = \begin{matrix} \text{symm.} \end{matrix}$$

$$A_2 = \begin{matrix} \text{symm.} \end{matrix}$$

$$A_3 = \begin{matrix} \text{symm.} \end{matrix}$$

↓ simultaneous block diagonalization

$$P^T A_1 P = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix}$$

$$P^T A_2 P = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix}$$

$$P^T A_3 P = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix}$$

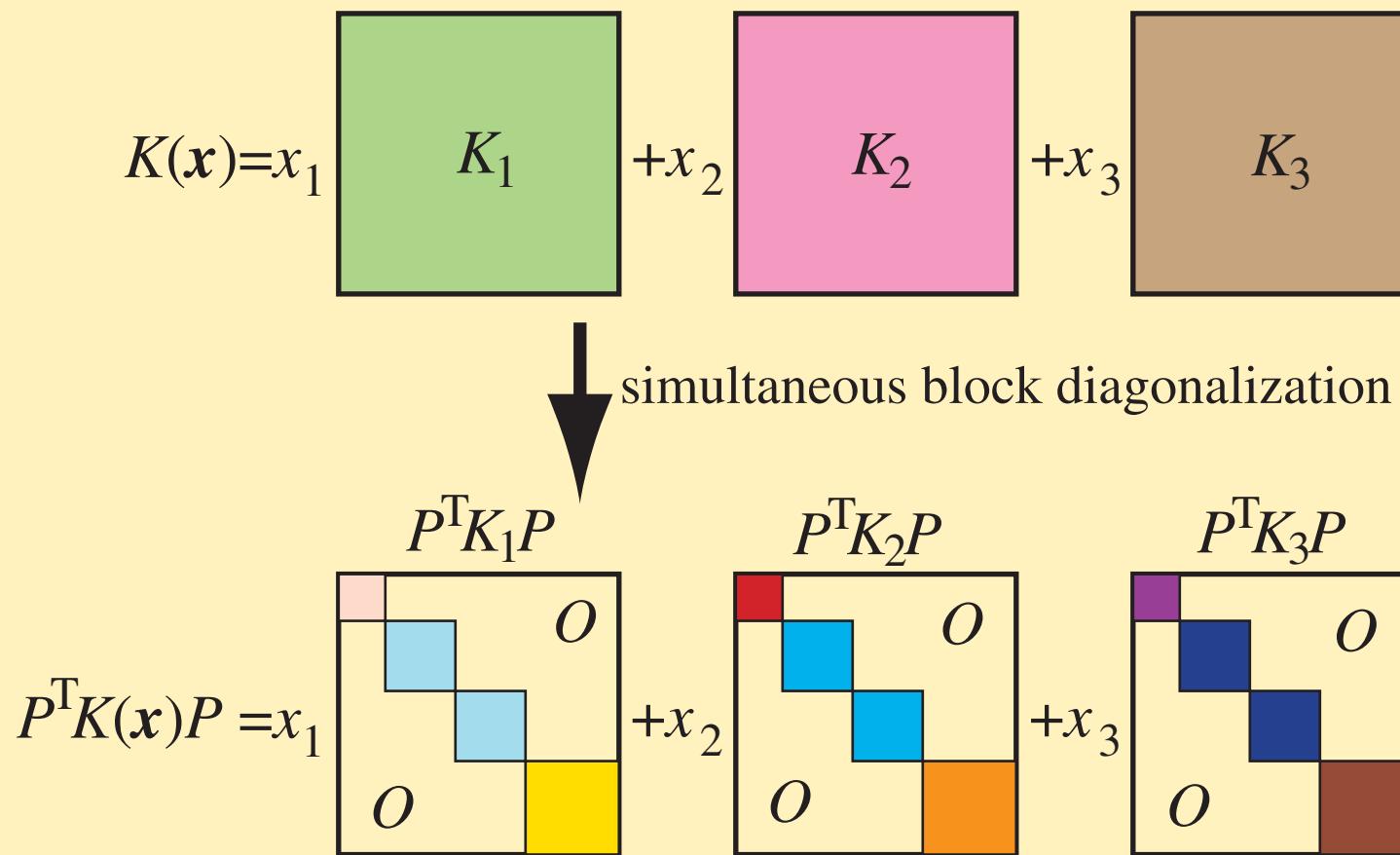
simultaneous block-diagonal decomposition: application

- $K(\boldsymbol{x})$: stiffness matrix \boldsymbol{x} : design variables
- given: K_1, K_2, \dots, K_m
- find: $P \in \mathbb{R}^{n \times n}$ — orthogonal

$$K(\boldsymbol{x}) = x_1 \begin{array}{|c|} \hline K_1 \\ \hline \end{array} + x_2 \begin{array}{|c|} \hline K_2 \\ \hline \end{array} + x_3 \begin{array}{|c|} \hline K_3 \\ \hline \end{array}$$

simultaneous block-diagonal decomposition: application

- $K(\mathbf{x})$: stiffness matrix \mathbf{x} : design variables
- given: K_1, K_2, \dots, K_m
- find: $P \in \mathbb{R}^{n \times n}$ — orthogonal



application: SDP

$$\begin{array}{ll}\min & \sum_{p=1}^m b_p y_p \\ \text{s.t.} & C - \sum_{p=1}^m A_p y_p \succeq O\end{array}$$

variables : y_1, \dots, y_m

coefficients : $b_1, \dots, b_m,$

$A_1, \dots, A_m, C \in \mathcal{S}^n$ $n \times n$ symmetric matrices

- $X \succeq O \iff X$ is positive semidefinite
← nonlinear but convex

application: SDP

$$\begin{array}{ll}\min & \sum_{p=1}^m b_p y_p \\ \text{s.t.} & C - \sum_{p=1}^m A_p y_p \succeq O\end{array}$$

- truss optimization (with eigenvalue constraints)

$$\begin{array}{ll}\min & vol(\mathbf{x}) \\ \text{s.t.} & \Omega_1(\mathbf{x}) \geq \bar{\Omega}\end{array}$$

can be solved by SDP [Ohsaki *et al.* 99] as

$$\begin{array}{ll}\min & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} & \sum_{p=1}^m (K_p - \bar{\Omega} M_p) y_p \succeq O\end{array}$$

- optimization with buckling load factor constraints
- robust structural optimization

application: SDP

$$\begin{array}{ll}\min & \sum_{p=1}^m b_p y_p \\ \text{s.t.} & C - \sum_{p=1}^m A_p y_p \succeq O\end{array}$$

$$P^T A_1 P = \begin{pmatrix} & & & O \\ & \text{blue} & \text{blue} & \\ & \text{blue} & & \\ O & & & \text{orange} \end{pmatrix} \quad P^T A_2 P = \begin{pmatrix} & & & O \\ & \text{purple} & \text{blue} & \\ & \text{blue} & \text{blue} & \\ O & & & \text{brown} \end{pmatrix} \dots \dots \quad P^T C P = \begin{pmatrix} & & & O \\ & \text{pink} & \text{light blue} & \\ & \text{light blue} & \text{light blue} & \\ O & & & \text{yellow} \end{pmatrix}$$

↓

$$C^{(1)} - \sum_{p=1}^m A_p^{(1)} y_p \succeq O$$

⋮

$$C^{(\ell)} - \sum_{p=1}^m A_p^{(\ell)} y_p \succeq O$$

application: SDP

$$\begin{array}{ll}\min & \sum_{p=1}^m b_p y_p \\ \text{s.t.} & C - \sum_{p=1}^m A_p y_p \succeq O\end{array}$$

$$P^T A_1 P = \begin{pmatrix} & & & O \\ & & O & \\ & O & & \\ O & & & \end{pmatrix} \quad P^T A_2 P = \begin{pmatrix} & & & O \\ & & O & \\ & O & & \\ O & & & \end{pmatrix} \dots \dots \quad P^T C P = \begin{pmatrix} & & & O \\ & & O & \\ & O & & \\ O & & & \end{pmatrix}$$

- group theory
[Gatermann & Parrilo 04], [Bai & de Klerk 07] [de Klerk & Sotirov 07]
- matrix $*$ -algebra [de Klerk, Pasechnik & Scrijver 07]
- aim: numerical algorithm
which can be executed **without** any knowledge
of symmetry property in advance

matrix *-algebra

$\mathcal{T} \subseteq \mathbb{R}^{n \times n}$ (set of $n \times n$ matrices)

is a matrix *-algebra



- $I_n \in \mathcal{T}$
- If $A, B \in \mathcal{T}$, $\alpha \in \mathbb{R}$, then
 - ◆ $A + B \in \mathcal{T}$
 - ◆ $AB \in \mathcal{T}$
 - ◆ $\alpha A \in \mathcal{T}$
 - ◆ $A^T \in \mathcal{T}$

structure theorem

1. $\exists Q$: orthogonal s.t.

$$Q^T \mathcal{T} Q = \begin{bmatrix} S_1 & O & \cdots & O \\ O & S_2 & O & O \\ \vdots & O & \ddots & O \\ O & O & \cdots & S_l \end{bmatrix}, \quad S_j \in \mathcal{T}_j : \text{simple}$$

2. If T_j is simple, then $\exists \hat{P}$: orthogonal s.t.

$$\hat{P}^T \mathcal{T}_j \hat{P} = \begin{bmatrix} B_j & O & O & O \\ O & B_j & O & O \\ O & O & \ddots & O \\ O & O & O & B_j \end{bmatrix}, \quad B_j \in \hat{\mathcal{T}}_j : \text{irreducible}$$

structure theorem: transformations to

- simple components

$\exists Q$: orthogonal, for any $A \in \mathcal{T}$,

$$Q^T A Q = \begin{matrix} S_1 & O \\ O & S_2 \end{matrix}$$

$Q^T A Q =$

no common eigenvalues

- & irreducible components

$\exists \hat{P}$: orthogonal, for any $A \in \mathcal{T}$,

structure theorem: transformations to

- simple components

$\exists Q$: orthogonal, for any $A \in \mathcal{T}$,

$$Q^T A Q = \begin{array}{|c|c|} \hline S_1 & O \\ \hline O & S_2 \\ \hline \end{array}$$

no common eigenvalues

$Q^T A Q =$

multiplicity = 3

multiplicity = 2

- & irreducible components

$\exists \hat{P}$: orthogonal, for any $A \in \mathcal{T}$,

$$P^T A P = \begin{array}{|c|c|} \hline B_1 & B_1 \\ \hline B_1 & B_1 \\ \hline \hline B_2 & B_2 \\ \hline B_2 & B_2 \\ \hline \end{array}$$

no multiplicity

no multiplicity

$P^T A P =$

algorithm (1)

input: A_1, A_2, \dots, A_m

1. Generate random r_1, \dots, r_m . Put $X \equiv \sum_{i=1}^m r_i A_i$.
2. Eigenvalue decomposition of X :

$$Q^T X Q = \begin{bmatrix} \alpha_1 I_{m_1} & O & O & O \\ \hline O & \alpha_2 I_{m_2} & O & O \\ \hline O & O & \ddots & O \\ \hline O & O & O & \alpha_k I_{m_k} \end{bmatrix},$$
$$Q = [Q_1, Q_2, \dots, Q_k].$$

3. Compute $Q^T A_1 Q, \dots, Q^T A_m Q$.

algorithm (1)

input: A_1, A_2, \dots, A_m

1. Generate random r_1, \dots, r_m . Put $X \equiv \sum_{i=1}^m r_i A_i$.
2. Eigenvalue decomposition of X :
3. Compute $Q^T A_1 Q, \dots, Q^T A_m Q$.
4. Rearrange eigenvectors Q as

$$\hat{Q} = [Q[K_1], Q[K_2], \dots, Q[K_l]]$$

so that

$$\hat{Q}^T A_1 \hat{Q} = \begin{array}{|c|c|} \hline \textcolor{pink}{\square} & O \\ \hline O & \textcolor{blue}{\square} \\ \hline \end{array} \quad \hat{Q}^T A_2 \hat{Q} = \begin{array}{|c|c|} \hline \textcolor{red}{\square} & O \\ \hline O & \textcolor{orange}{\square} \\ \hline \end{array} \dots$$

which is the decomposition to simple components.

algorithm (2)

input: $\hat{A}_1, \hat{A}_2, \dots, \hat{A}_m$ — simple

1. We know that the simple component is decomposed as

$$P^T \hat{A} P = \left. \begin{array}{c|c|c} B & & \\ \hline & B & \\ \hline & & B \end{array} \right\} \text{multiplicity} = 3$$

2. e.g. $B \in \mathbb{R}^{2 \times 2}$, $P^T \hat{A} P$ can be rearranged as

$$\tilde{P}^T \hat{A} \tilde{P} = \left[\begin{array}{c|c|c} b_{11}I & b_{12}I & b_{13}I \\ \hline b_{21}I & b_{22}I & b_{23}I \\ \hline b_{31}I & b_{32}I & b_{33}I \end{array} \right], \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (\clubsuit)$$

We first find the transformation (\clubsuit) .

algorithm (2)

input: $\hat{A}_1, \hat{A}_2, \dots, \hat{A}_m$ — simple

1. In order to obtain

$$\tilde{P}^T \hat{A} \tilde{P} = \left[\begin{array}{c|c|c} b_{11}I & b_{12}I & b_{13}I \\ \hline b_{21}I & b_{22}I & b_{23}I \\ \hline b_{31}I & b_{32}I & b_{33}I \end{array} \right], \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (\clubsuit)$$

we use the form of

$$b_{ij} P_i P_j^T = \hat{A}_{ij}.$$

2. Put $P_1 = I$. Then,

$$b_{31} P_3 P_1^T = \hat{A}_{31} \implies P_3 = \hat{A}_{31}/b_{31}$$

$$b_{23} P_2 P_3^T = \hat{A}_{23} \implies P_2 = \hat{A}_{23} P_3 / b_{23}$$

algorithm (1) & (2)

- (1) simple components

$$\hat{Q}^T A_1 \hat{Q} = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix} \quad \hat{Q}^T A_2 \hat{Q} = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix} \dots$$

- (2) irreducible components

$$P^T A_1 P = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix} \quad P^T A_2 P = \begin{matrix} & & O \\ & O & \\ O & & \end{matrix} \dots$$

algorithm (1) & (2)

■ (1) simple components

$$\hat{Q}^T A_1 \hat{Q} = \begin{pmatrix} O & & \\ & O & \\ & & O \end{pmatrix} \quad \hat{Q}^T A_2 \hat{Q} = \begin{pmatrix} O & & \\ & O & \\ & & O \end{pmatrix} \dots$$

■ (2) irreducible components

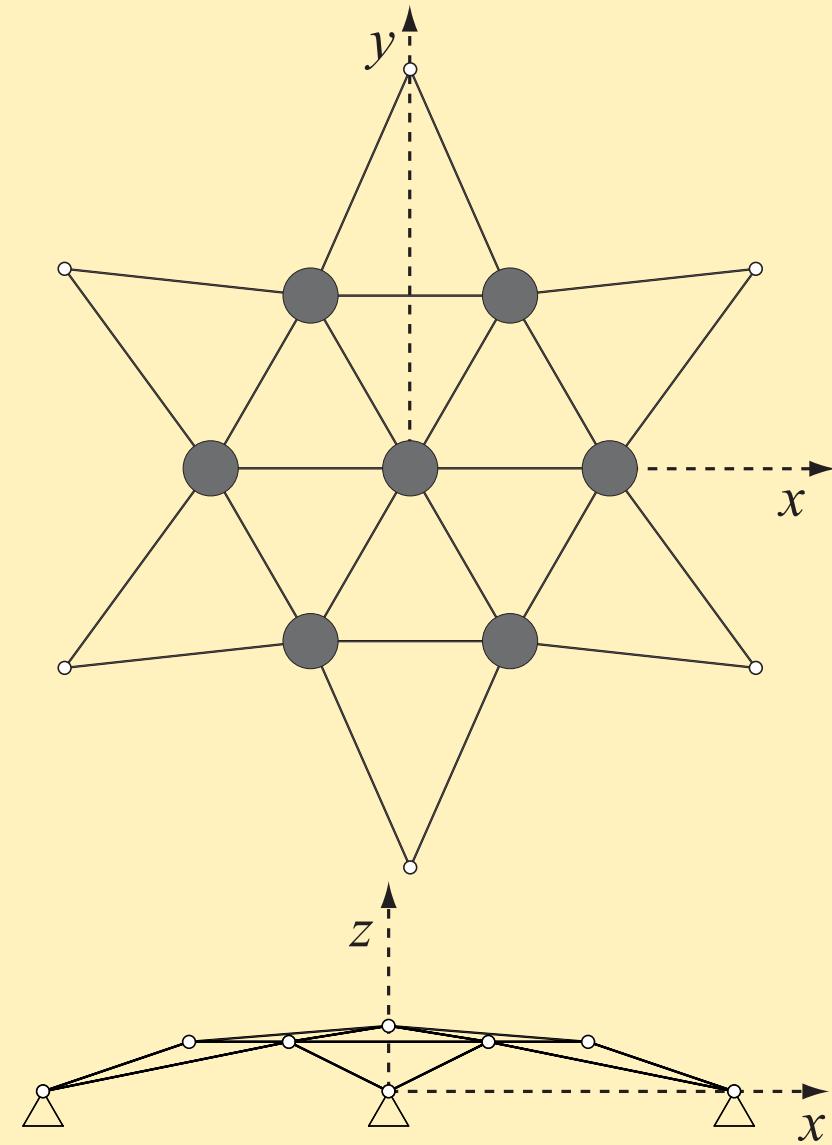
$$P^T A_1 P = \begin{pmatrix} O & & \\ & O & \\ & & O \end{pmatrix} \quad P^T A_2 P = \begin{pmatrix} O & & \\ & O & \\ & & O \end{pmatrix} \dots$$

■ advantages: algorithm uses

- ◆ only **linear algebraic computations** (e.g. eigenvalue decomposition)
- ◆ no knowledge of **algebraic structure** (**symmetry property**)
execution is free from **group representation theory**
or **matrix *-algebra**

ex.) spherical dome

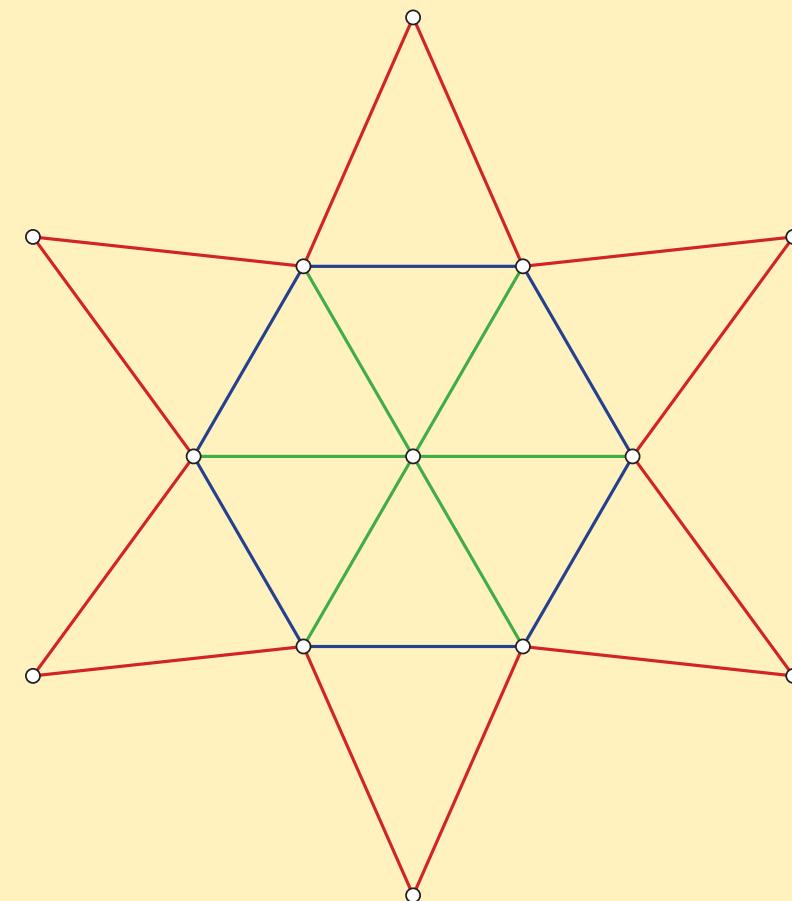
- spherical lattice dome
 D_6 -symmetry



ex.) spherical dome

- spherical lattice dome
- D₆-symmetry
- variable linking
- 3 kinds of members
- $K(\mathbf{x}) =$
 $x_1 K_1 + x_2 K_2 + x_3 K_3$
- 21 DOF

$$K_i = \underbrace{\begin{matrix} \text{symm.} \end{matrix}}_{21} \quad \left. \right\} 21$$



ex.) spherical dome

- spherical lattice dome

D_6 -symmetry

- variable linking

3 kinds of members

$$K(\mathbf{x}) =$$

$$x_1 K_1 + x_2 K_2 + x_3 K_3$$

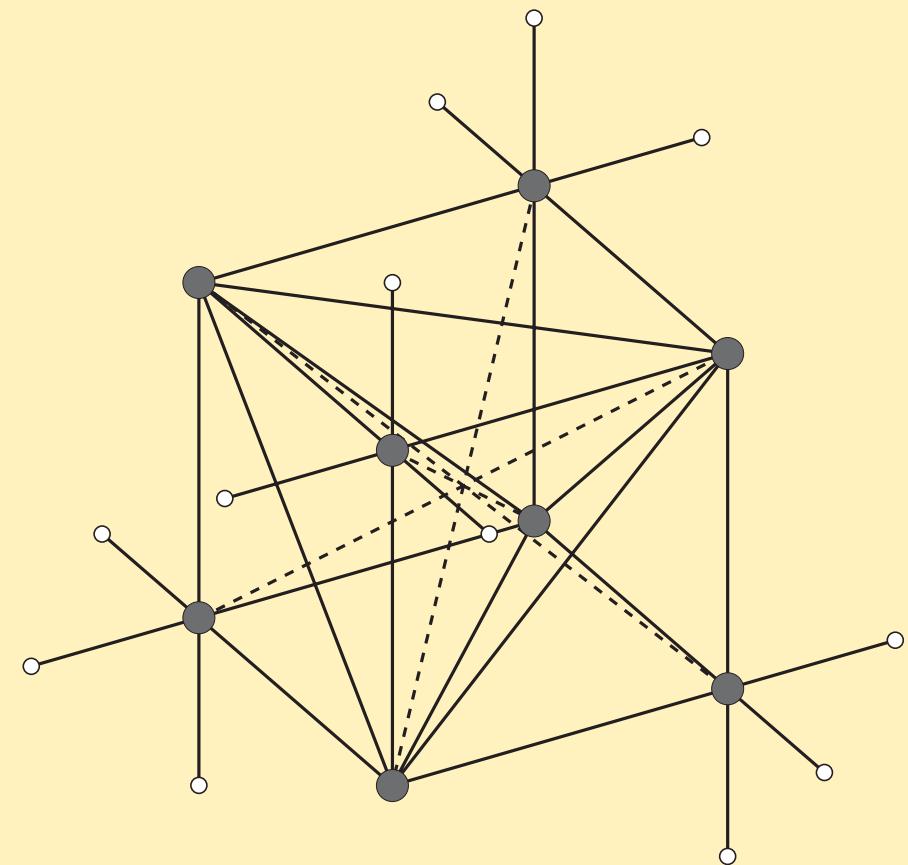
- 21 DOF

$$K_i = \underbrace{\begin{matrix} \text{symm.} \end{matrix}}_{21} \quad \left. \right\} 21$$

$$P^T K_i P =$$

ex.) cubic truss (tetrahedral symm.)

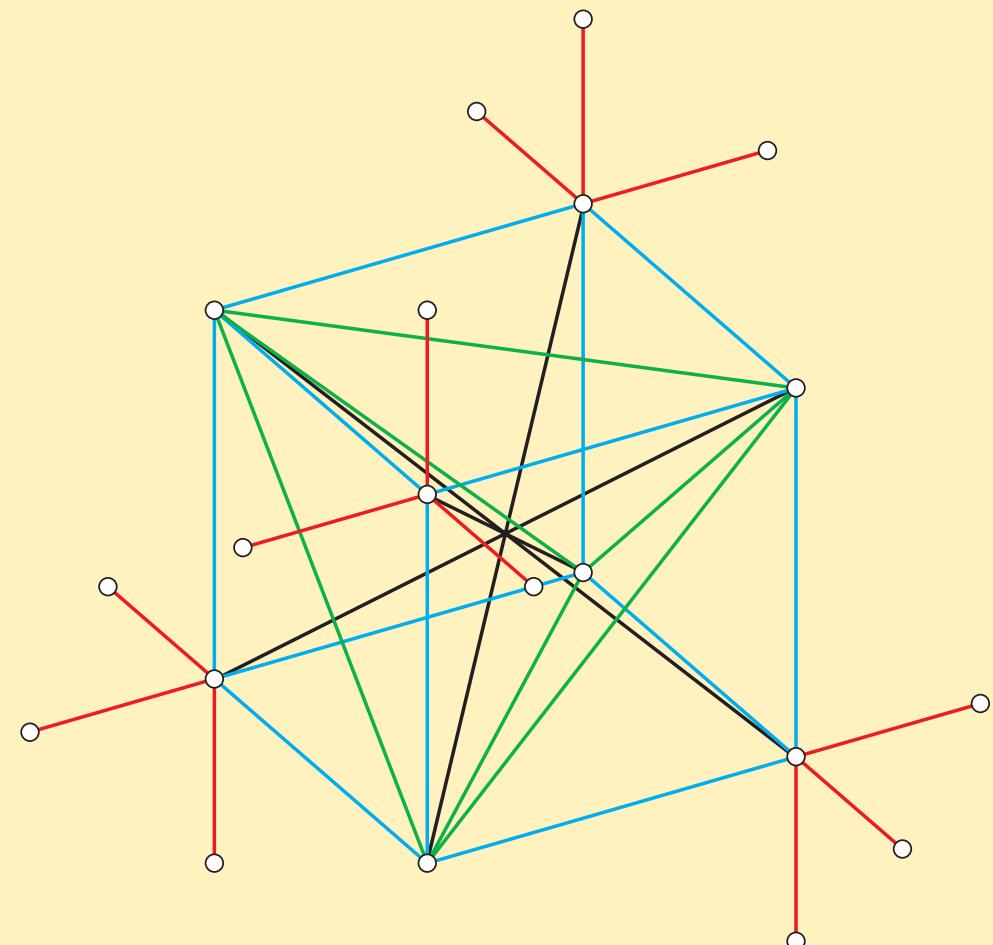
- T_d -symmetry (tetrahedral)



ex.) cubic truss (tetrahedral symm.)

- T_d -symmetry (tetrahedral)
- variable linking
- 4 kinds of members
- $K(\mathbf{x}) = x_1 K_1 + \dots + x_4 K_4$
- 24 DOF

$$K_i = \underbrace{\begin{array}{|c|} \hline \text{symm.} \\ \hline \end{array}}_{24} \quad \left. \right\} 24$$



ex.) cubic truss (tetrahedral symm.)

- T_d -symmetry (tetrahedral)
- variable linking
- 4 kinds of members
- $K(\mathbf{x}) = x_1 K_1 + \dots + x_4 K_4$
- 24 DOF

$$K_i = \begin{matrix} \text{symm.} \\ \underbrace{\hspace{10em}}_{24} \end{matrix} \quad \left. \right\} 24$$

$$P^T K_i P = \begin{matrix} O & & & \\ & O & & \\ & & \overbrace{\hspace{1.5em}}^{2} & \overbrace{\hspace{1.5em}}^{2} & \overbrace{\hspace{3.5em}}^{4} \\ & & & & \end{matrix}$$

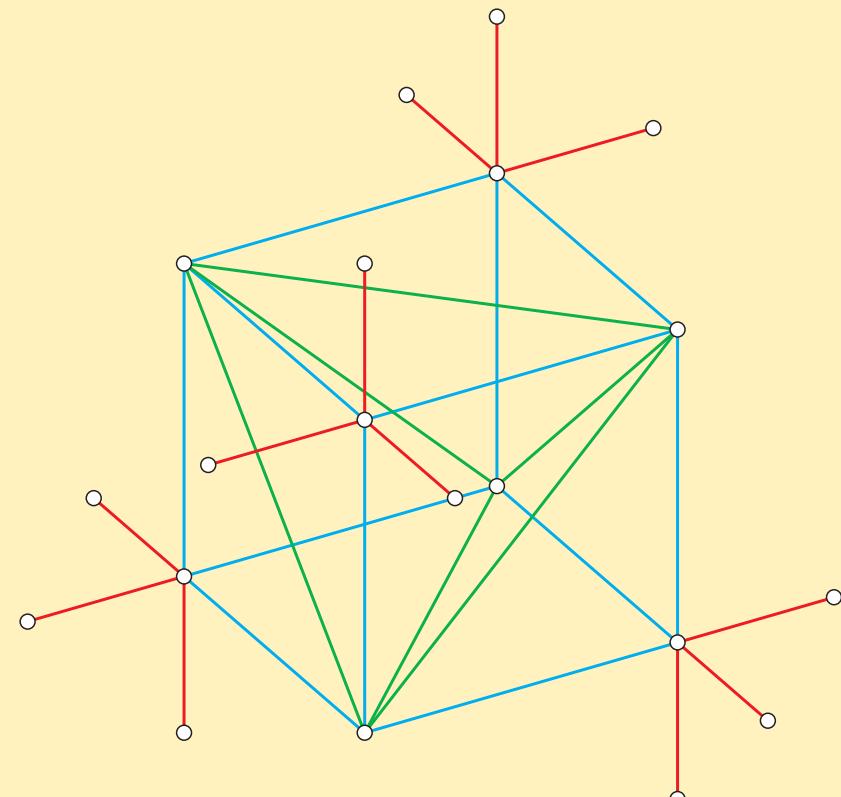
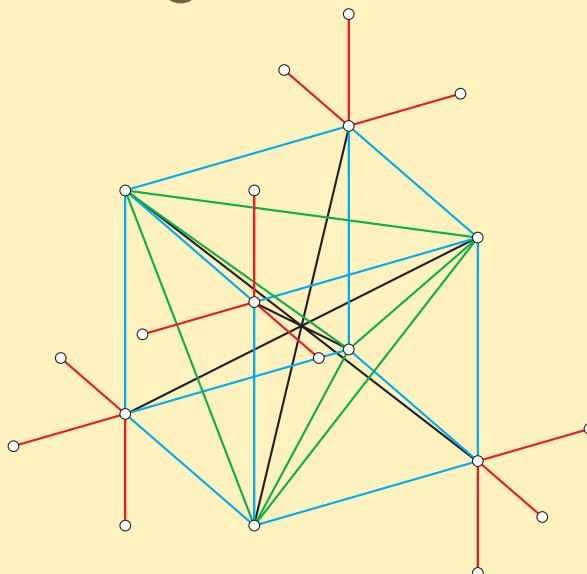
ex.) cubic truss (sparsity)

- remove 4 members
- variable linking
- 3 kinds of members

$$K(\mathbf{x}) = x_1 K_1 + \cdots + x_3 K_3$$

- 24 DOF

cf. original:



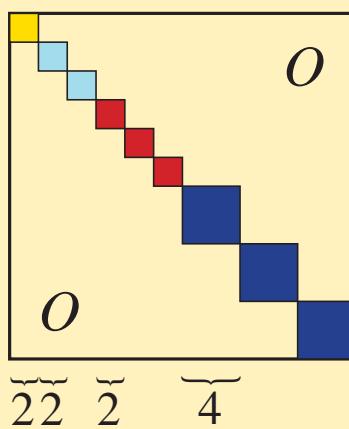
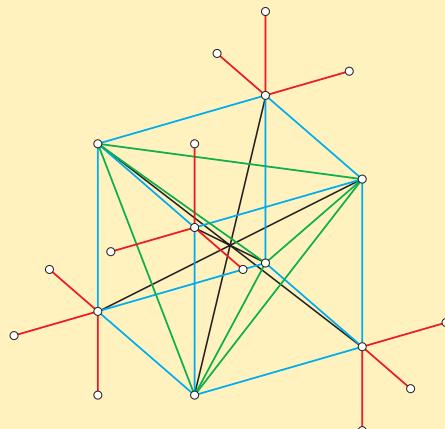
ex.) cubic truss (sparsity)

- remove 4 members
- variable linking
- 3 kinds of members

$$K(\mathbf{x}) = x_1 K_1 + \cdots + x_3 K_3$$

- 24 DOF

cf. original:



block-diagonal decomposition
= geometric symmetry
+ sparsity of matrices

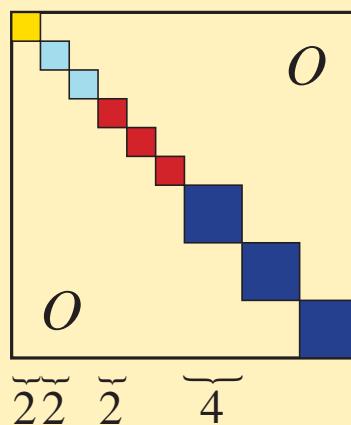
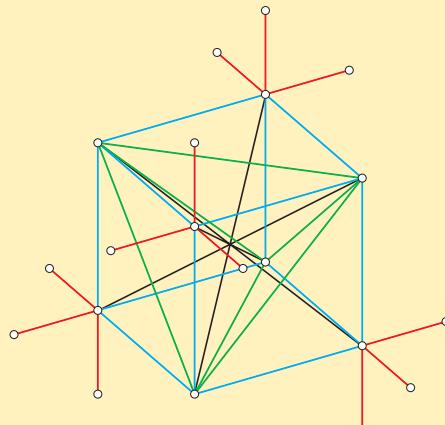
ex.) cubic truss (sparsity)

- remove 4 members
- variable linking
- 3 kinds of members

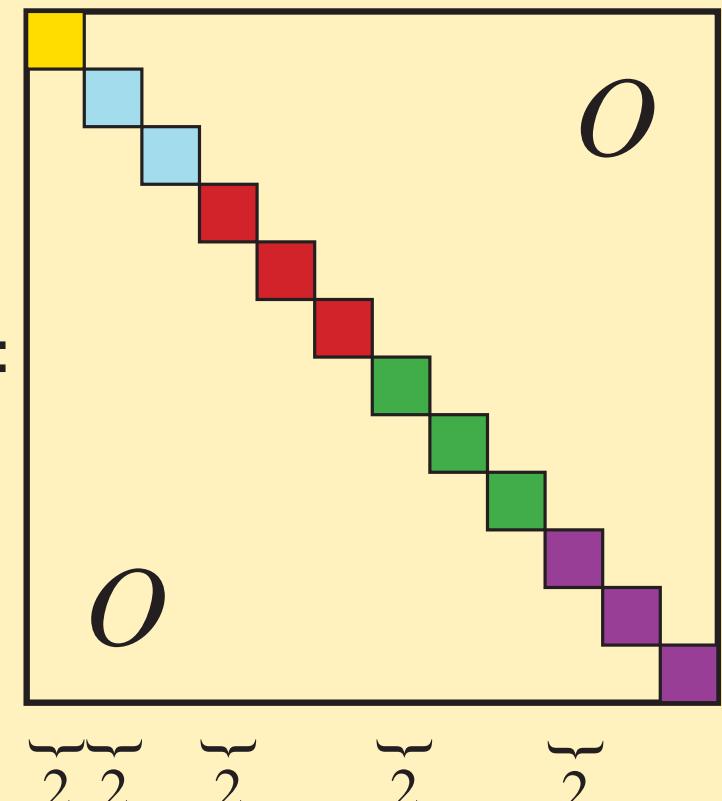
$$K(\mathbf{x}) = x_1 K_1 + \cdots + x_3 K_3$$

- 24 DOF

cf. original:



$$P^T K_i P =$$



sparsity gives finer decomposition

conclusions

■ simultaneous block-diagonalization

- ◆ input: A_1, A_2, \dots, A_m : symmetric matrices
- ◆ output: block-diagonal decomposition (common block-structure)

- ◆ validity: matrix $*$ -algebra
- ◆ application: member stiffness matrices of structure with
symmetric configuration

■ algorithm

- ◆ uses only linear algebraic computation
- ◆ does not use any knowledge of symmetry property
- ◆ is free from
 - group representation theory / matrix $*$ -algebra