

Florian Hüter, Johannes Wittmann

Titel: Statische Reduktion von FE-Strukturen nach Guyan – Effizienzsteigerung durch den Schur-Komplement-Ansatz

Abstract:

Der effiziente Einsatz von Simulationswerkzeugen im Produktentwicklungsprozess und damit die Einsparung von Werkstoffen, Fertigungs- und Entwicklungszeit gewinnen zunehmend an Bedeutung. Mehrkörpersimulationen (MKS) ermöglichen die Vorhersage des Verformungsverhaltens komplexer Baugruppen. Dabei werden die einzelnen FE-Komponenten der Baugruppe auf die Anschlussstellen (Master-Freiheitsgrade) zu den angrenzenden Komponenten reduziert, indem die übrigen Freiheitsgrade (Slave-Freiheitsgrade) aus dem FE-Gleichungssystem eliminiert werden, um so den Berechnungsaufwand für das Lösen des Gesamtsystems möglichst gering zu halten. Insbesondere bei der Betrachtung mehrerer Betriebspunkte kann so eine recheneffiziente Mehrfachlösung des Gesamtsystems erreicht werden.

Ein etablierter Ansatz zur Modellreduktion ist das statische Reduktionsverfahren nach Guyan. Auch wenn das Verfahren mathematisch eindeutig ist, so lassen sich für seine programmiertechnische Umsetzung verschiedene Algorithmen definieren, die sich im Hinblick auf die Zwischenschritte und ihre Effizienz (Rechenzeit, Arbeitsspeicher) unterscheiden. Im Rahmen dieses Beitrags werden zwei unterschiedliche Algorithmen vorgestellt und miteinander verglichen.

Algorithmus 1 basiert im ersten Schritt auf der Bildung einer Transformationsmatrix, durch deren Multiplikation mit dem FE-Gleichungssystem im zweiten Schritt die statische Reduktion auf die Master-Freiheitsgrade erfolgt. Die Transformationsmatrix entspricht dabei der Recovery-Matrix, die zur Berechnung der Verschiebungen an den Slave-Freiheitsgraden benötigt wird. Nachteilig ist jedoch, dass die Recovery-Matrix im Gegensatz zur Steifigkeitsmatrix nicht dünn, sondern voll besetzt ist, so dass der Rechen- und Speicheraufwand mit steigender Anzahl der Master-Freiheitsgrade deutlich zunimmt und der Algorithmus bei FE-Strukturen mit vielen Freiheitsgraden häufig nicht mehr anwendbar ist.

Dagegen basiert Algorithmus 2 auf der Bildung des Schur-Komplements der Steifigkeitsmatrix durch LR-Zerlegung, wobei das Schur-Komplement genau der reduzierten Steifigkeitsmatrix entspricht. Auf diese Weise wird der explizite Aufbau der Recovery-Matrix umgangen und somit Rechenzeit und Arbeitsspeicher eingespart, weshalb sich Algorithmus 2 insbesondere für die Reduktion großer FE-Strukturen eignet. Durch die fehlende Recovery-Matrix ist jedoch die Berechnung der Verschiebung an den Slave-Freiheitsgraden nur durch Lösung des FE-Gleichungssystems möglich, was jedoch nur einen geringen Mehraufwand bedeutet, wenn die Dreiecksmatrizen aus der LR-Zerlegung zwischengespeichert werden.

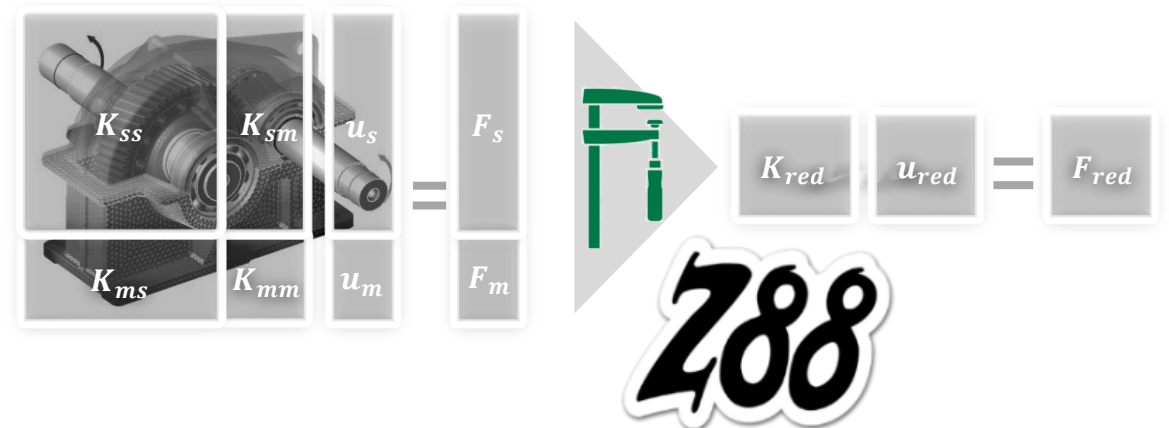
Anhand eines Berechnungsbeispiels mit Z88 zeigt dieser Beitrag die höhere Rechen- und Speichereffizienz von Algorithmus 2.

Statische Reduktion von FE-Strukturen nach GUYAN – Effizienzsteigerung durch den Schur-Komplement-Ansatz

Florian Hüter, Johannes Wittmann

25. Bayreuther 3D-Konstrukteurstag

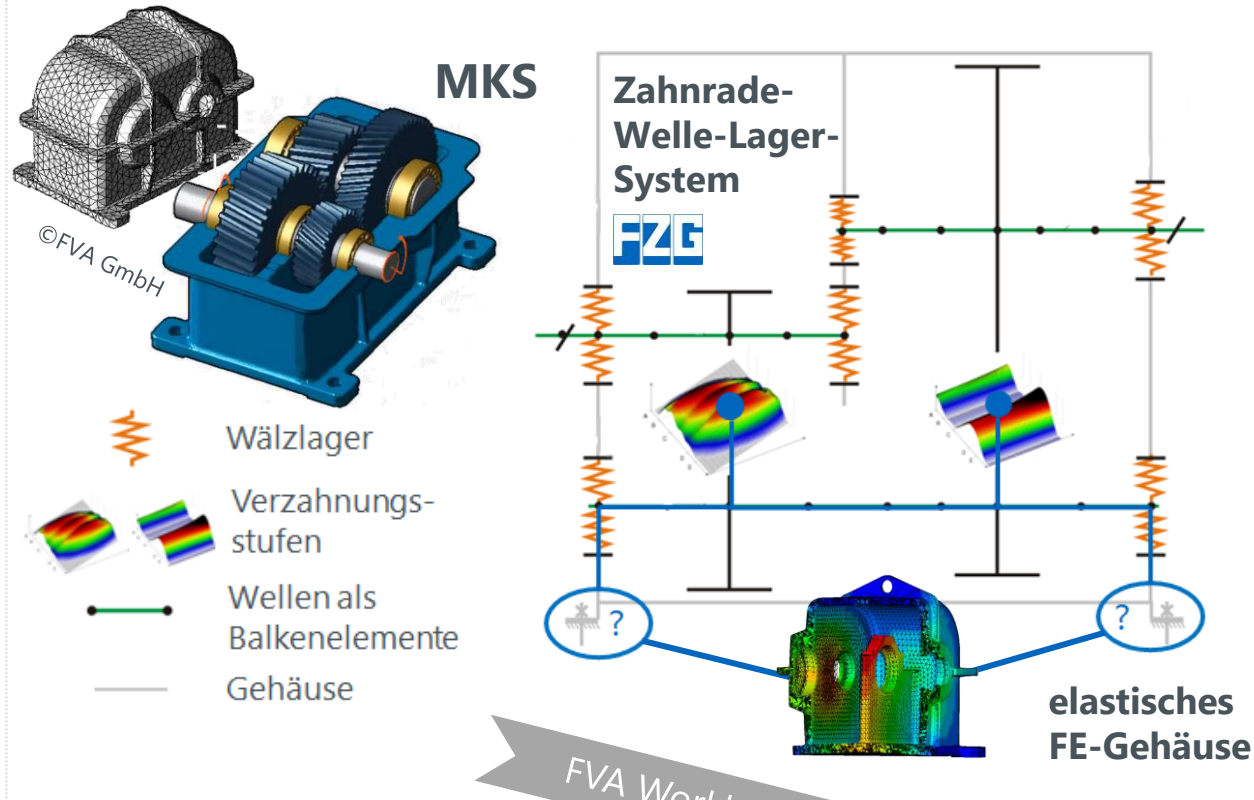
Bayreuth, 11.09.2024



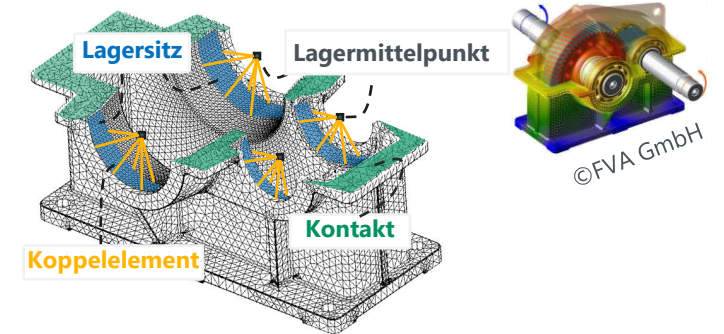
Statische Reduktion von FE-Strukturen

Motivation: Reduktion elastischer Körper für MKS

Quasi-statische Gesamtsystem-Simulation mit RIKOR



Reduktion Gehäusestruktur



- recheneffiziente Berücksichtigung des elastischen Gehäuseeinflusses durch Reduktion der FE-Gehäusestruktur auf Freiheitsgrade der Lagersitze

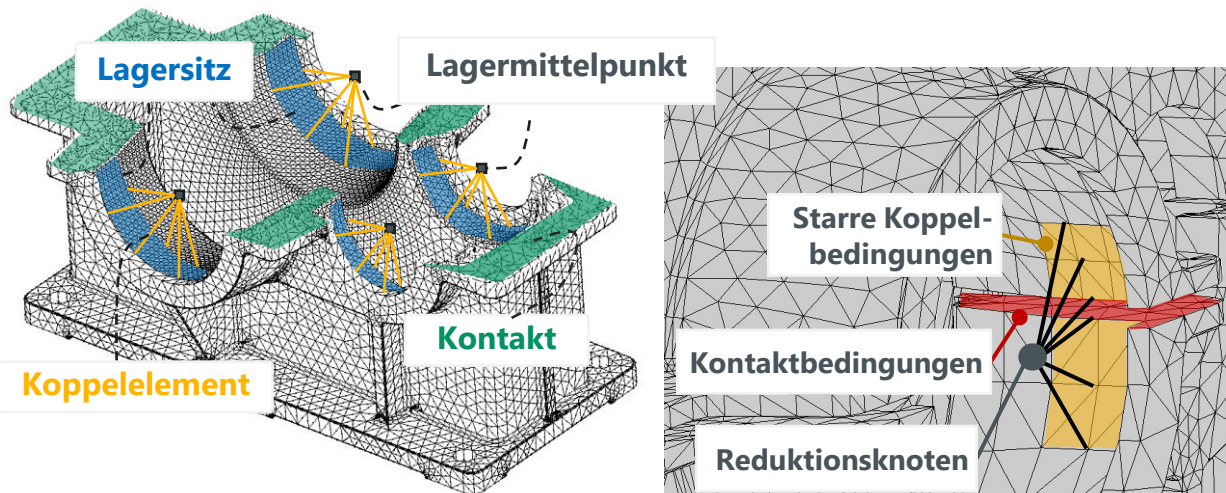
$$\begin{bmatrix} K_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} \begin{bmatrix} u_s \\ u_m \end{bmatrix} = \begin{bmatrix} F_s \\ F_m \end{bmatrix} \Rightarrow \begin{bmatrix} K_{red} \\ u_{red} \end{bmatrix} = \begin{bmatrix} F_{red} \end{bmatrix}$$



Statische Reduktion von FE-Strukturen

Statische Reduktion mit kinematischen Zwangsbedingungen

Reduktion von geteilten FE-Gehäusen auf Lagersitze



FE-Struktur mit kinematischen Zwangsbedingungen

- Kontaktbedingungen
- Koppelbedingungen

FE-Gleichungssystem

■ FE-Gleichungssystem $K \cdot u = F$

■ Zwangsbedingungen $G \cdot u = g$

➔ Zusammenbau zum FE-Gesamtsystem:

LAGRANGE-Methode

$$\begin{bmatrix} K & G^T \\ G & -\frac{1}{\beta} I \end{bmatrix} \cdot \begin{pmatrix} u \\ \lambda \end{pmatrix} = \begin{pmatrix} F \\ g \end{pmatrix}$$

Straf-
parameter

Penalty-Methode

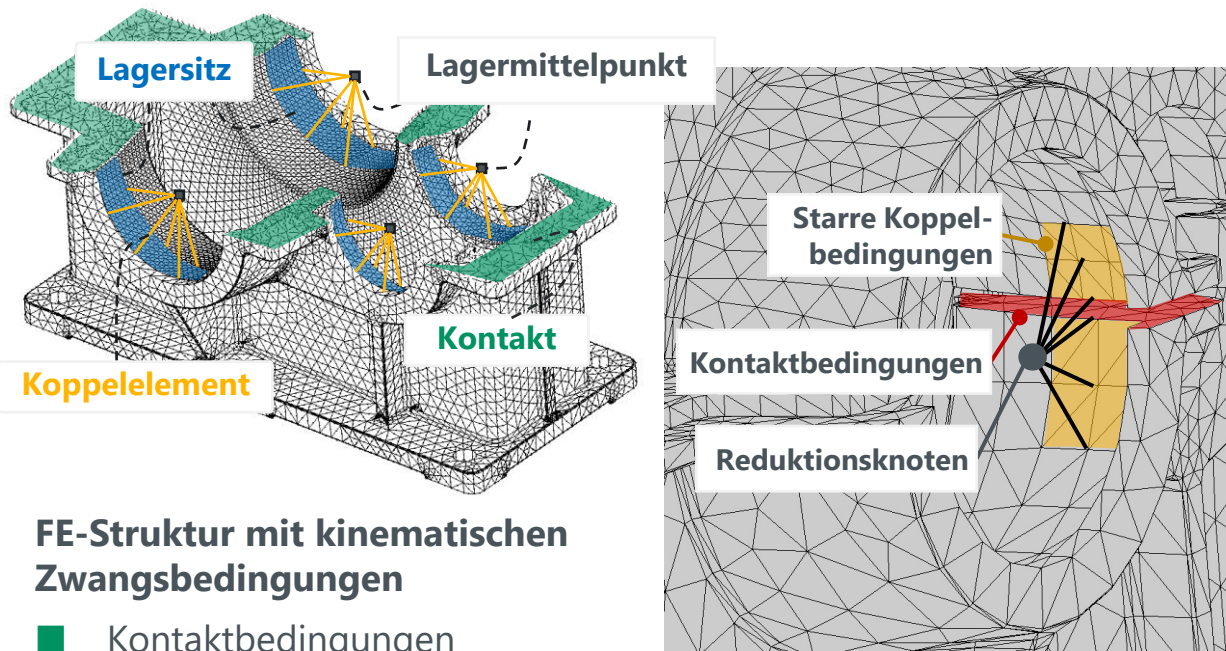
$$[K + \beta G^T G] \cdot u = F + \beta G^T g$$



Statische Reduktion von FE-Strukturen

Statische Reduktion mit kinematischen Zwangsbedingungen

Reduktion von geteilten FE-Gehäusen auf Lagersitze



FE-Struktur mit kinematischen Zwangsbedingungen

- Kontaktbedingungen
- Koppelbedingungen

FE-Gleichungssystem

■ FE-Gleichungssystem $K \cdot u = F$

■ Zwangsbedingungen $G \cdot u = g$

➔ Zusammenbau zum FE-Gesamtsystem:

Strafparameter bestimmen

- Globales Kriterium:

$$\beta = \beta_{rel} \cdot \max(K_{ii})$$

- Lokales Kriterium:

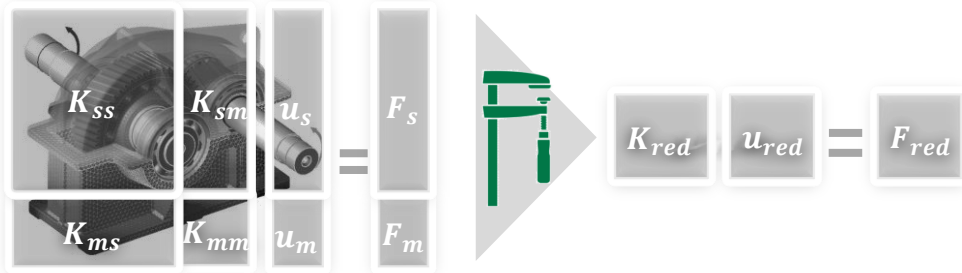
$$\beta = \beta_{rel} \cdot 10^{-3} \cdot \frac{\min(K_{ii})}{\sqrt{N_{dim} \cdot t}}$$

$$\beta_{rel} \approx 10^{3...4}$$



Idee der statischen Reduktion

- **Ziel:** FE-Gleichungssystem auf relevante Freiheitsgrade (=Master-FG) reduzieren



- **Ansatz:** Elimination der irrelevanten Freiheitsgrade (=Slave-FG) aus dem Gleichungssystem
- **Algorithmen:**
 - Direktelimination
 - Transformationsmatrix-Ansatz
 - SCHUR-Komplement-Ansatz

Direktelimination d. Slave-FG

- **Partitionierung** in Master-FG und Slave-FG:

$$\text{FE-System} \quad \begin{bmatrix} K_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} \begin{pmatrix} u_s \\ u_m \end{pmatrix} = \begin{pmatrix} F_s \\ F_m \end{pmatrix}$$

$$\Rightarrow u_s = K_{ss}^{-1} F_s - K_{ss}^{-1} K_{sm} u_m$$

- **Elimination** der Slave-Freiheitsgrade (Slave-FG):

**Reduziertes
FE-System**

$$K_{red} u_m = F_{red}$$

mit

$$K_{red} := K_{mm} - K_{ms} K_{ss}^{-1} K_{sm}$$

$$F_{red} := F_m - K_{ms} K_{ss}^{-1} F_s$$

- *Steifigkeitsmatrix ist dünnbesetzt*
- *Reduzierte Matrix ist dichtbesetzt wegen Inverse*



Recovery des Gesamtverschiebungsfeldes

$$\begin{pmatrix} \mathbf{u}_s \\ \mathbf{u}_m \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{ss}^{-1} \mathbf{F}_s - \mathbf{K}_{ss}^{-1} \mathbf{K}_{sm} \mathbf{u}_m \\ \mathbf{u}_m \end{pmatrix}$$

$$= \underbrace{\begin{bmatrix} -\mathbf{K}_{ss}^{-1} \mathbf{K}_{sm} \\ \mathbf{I} \end{bmatrix}}_{\mathbf{T}} \mathbf{u}_m + \begin{bmatrix} \mathbf{K}_{ss}^{-1} \mathbf{F}_s \\ \mathbf{0} \end{bmatrix}$$

Transformationsmatrix \mathbf{T}

Reduktion mittels Transformationsmatrix

$$\delta \mathbf{u}^T (\mathbf{K} \mathbf{u} - \mathbf{F}) = 0 \rightarrow \delta \mathbf{u}_m^T \mathbf{T}^T (\mathbf{K} \mathbf{T} \mathbf{u}_m - \mathbf{F}) = 0$$

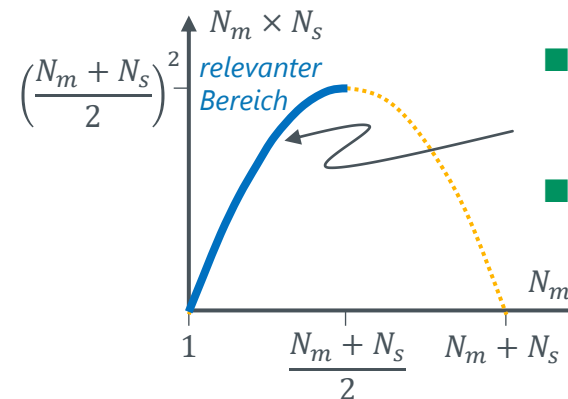
$$\underbrace{\mathbf{T}^T \mathbf{K} \mathbf{T}}_{\mathbf{K}_{red}} \cdot \mathbf{u}_m = \underbrace{\mathbf{T}^T \mathbf{F}}_{\mathbf{F}_{red}}$$

Bestimmung der Transformationsmatrix

- \mathbf{K}_{ss}^{-1} nicht explizit durch Invertierung ermitteln
- effizienter Ansatz mittels GAUß-JORDAN-Verfahren:

$$\begin{bmatrix} \mathbf{K}_{ss} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{pmatrix} \mathbf{T}_s \\ \mathbf{T}_m \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{sm} \\ \mathbf{I} \end{pmatrix}$$

Bestimmung der Transformationsmatrix



- Speicheraufwand $N_m \times N_s$ steigt mit zunehmender Anzahl an Master-FG N_m
- und damit die Anzahl zu berechnender Matrixeinträge

Verfahren für große N_m sehr aufwendig!



Matrix-Reduktion durch LR-Zerlegung

- **Idee:** unvollständige Matrixfaktorisierung
 - Direct Sparse Matrix Solver basieren auf einer Matrix-Faktorisierung (LR, LDL, CHOLESKY)
 - Führt man die Faktorisierung der Matrix nur für die Slave-FG durch, so entspricht der noch unfaktorisierte Matrixteil (=SCHUR-Komplement S) genau der reduzierten Matrix
- **Exemplarisch:** unvollständige LR-Zerlegung

$$\begin{aligned}
 \begin{bmatrix} K_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} &= \begin{bmatrix} L_{ss}R_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} \\
 &= \begin{bmatrix} L_{ss} & 0 \\ K_{ms}R_{ss}^{-1} & S \end{bmatrix} \cdot \begin{bmatrix} R_{ss} & L_{ss}^{-1}K_{sm} \\ 0 & I \end{bmatrix}
 \end{aligned}$$

SCHUR-Komplement $S \equiv K_{red} = K_{mm} - K_{ms}K_{ss}^{-1}K_{sm}$

Reduzierter Kraftvektor

- (1) Vorwärts-Einsetzen: $L_{ss}X = F_s \rightarrow X$
- (2) Matrix-Vektor-Produkt: $(K_{ms}R_{ss}^{-1}) \cdot X \rightarrow Y$
- (3) Vektor-Subtraktion: $F_m - Y \rightarrow F_{red}$

Recovery Verschiebungsfeld

- (1) Matrix-Vektor-Produkt: $(L_{ss}^{-1}K_{sm}) \cdot u_m \rightarrow X$
- (2) Vorwärts-Einsetzen: $L_{ss}Y = F_s \rightarrow Y$
- (3) Rückwärts-Einsetzen:

$$R_{ss}[\tilde{X} \quad \tilde{Y}] = [X \quad Y] \rightarrow [\tilde{X} \quad \tilde{Y}]$$
- (4) Vektor-Subtraktion:

$$\tilde{Y} - \tilde{X} \rightarrow u_s$$



- Hauptrechenaufwand für Matrixfaktorisierung
- Speichereffiziente Sparse-Matrix-Speicherung



Matrix-Reduktion durch LR-Zerlegung

- **Idee:** unvollständige Matrixfaktorisierung
 - Direct Sparse Matrix Solver basieren auf einer Matrix-Faktorisierung (LR, LDL, CHOLESKY)
 - Führt man die Faktorisierung der Matrix nur für die Slave-FG durch, so entspricht der noch unfaktorierte Matrixteil (=SCHUR-Komplement S) genau der reduzierten Matrix
- **Exemplarisch:** unvollständige LR-Zerlegung

$$\begin{aligned} \begin{bmatrix} K_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} &= \begin{bmatrix} L_{ss}R_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix} \\ &= \begin{bmatrix} L_{ss} & 0 \\ K_{ms}R_{ss}^{-1} & S \end{bmatrix} \cdot \begin{bmatrix} R_{ss} & L_{ss}^{-1}K_{sm} \\ 0 & I \end{bmatrix} \end{aligned}$$

SCHUR-Komplement $S \equiv K_{red} = K_{mm} - K_{ms}K_{ss}^{-1}K_{sm}$

Reduzierter Kraftvektor

- (1) Vorwärts-Einsetzen: $L_{ss}X = F_s \rightarrow X$
- (2) Matrix-Vektor-Produkt: $(K_{ms}R_{ss}^{-1}) \cdot X \rightarrow Y$
- (3) Vektor-Subtraktion: $F_m - Y \rightarrow F_{red}$

Eigenschaften

- kein Aufbau der Transformationsmatrix
→ rechenzeit- und speicherintensivster Schritt aus Algorithmus 1 entfällt hier!
- Hauptrechenaufwand für Matrixfaktorisierung von K_{ss} auch in Algorithmus 1 vorhanden
- Unvollständige Matrixfaktorisierung mit bekannten Sparse Direct Solver (z. B. PARDISO) möglich



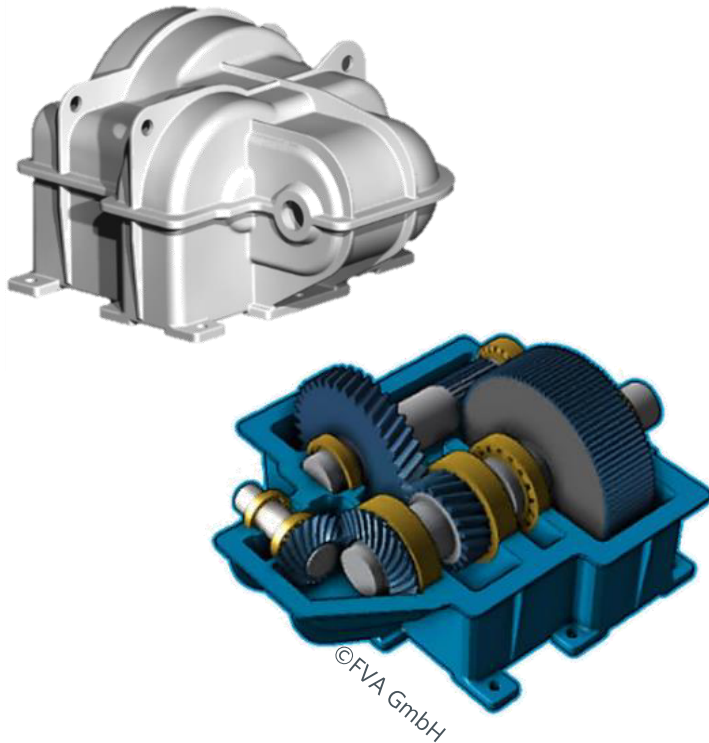
**rechen- & speichereffizienter,
insbesondere bei großen N_m**



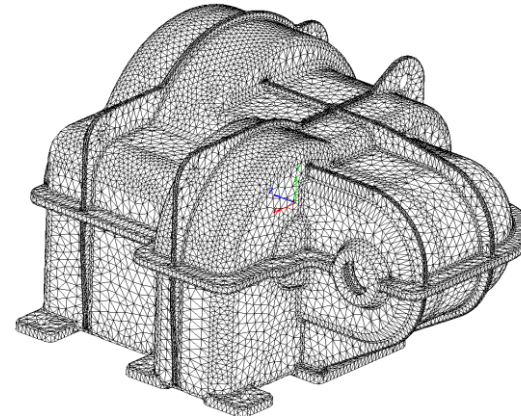
Vergleich der Reduktionsalgorithmen

Anwendungsbeispiel: Getriebegehäuse

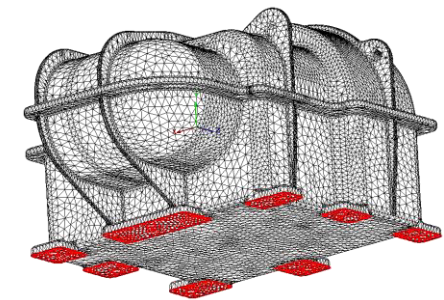
3-stufiges Getriebe



Statische Reduktion auf Lagerstellen



- Festhaltung an Fundamentanbindung



Modelleigenschaften

- GG-Gehäuse mit 8 Lagerstellen
- 218.134 quad. Tet.
- 1.151.328 Freiheitsgrade

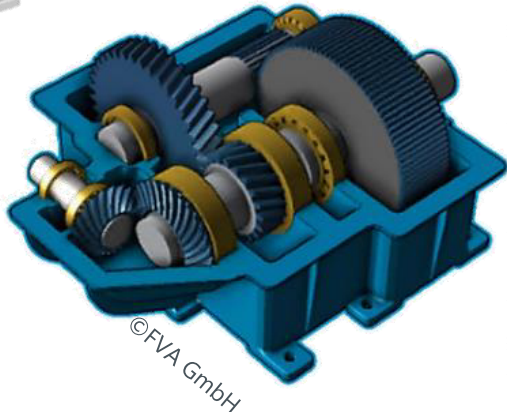
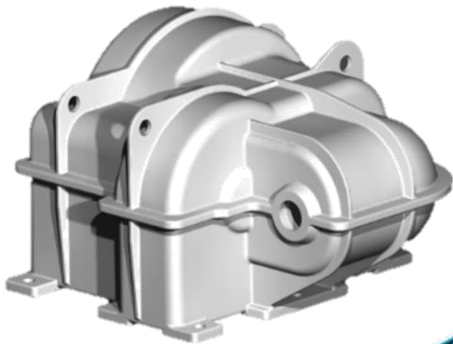
Z88



Vergleich der Reduktionsalgorithmen

Anwendungsbeispiel: Getriebegehäuse

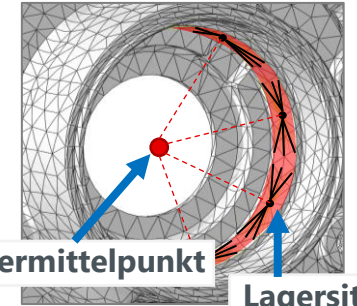
3-stufiges Getriebe



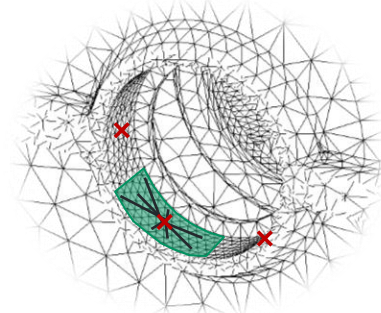
Statische Reduktion auf Lagerstellen

■ Modellierung der Gehäuse-Lager-Anbindung

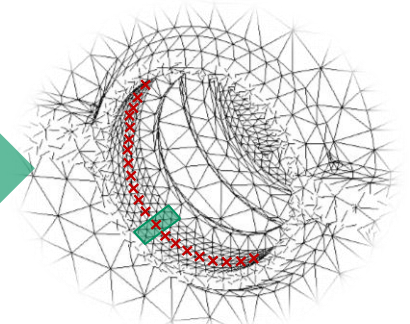
- Modellierungsstrategie zur Abbildung der Lagerovalisierung nach FVA 711 II
- Unterteilung in Lagersitzsegmente
- Starre Koppelbedingung zwischen Segment-Knoten & Reduktionsknoten



➔ Variation der Anzahl der Lagersitzsegmente



schrittweise Erhöhung
der Segmentanzahl



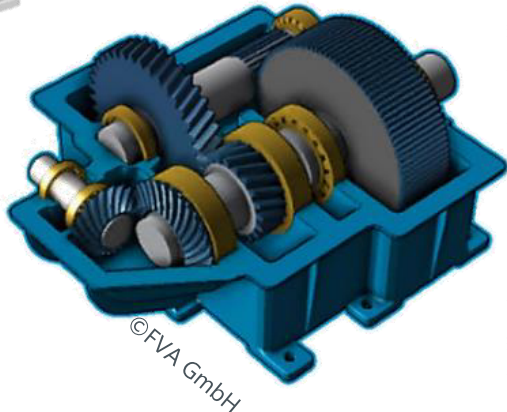
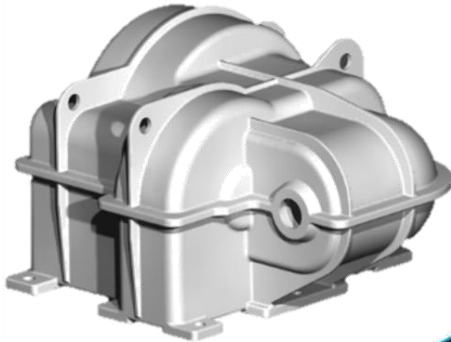
Z88



Vergleich der Reduktionsalgorithmen

Anwendungsbeispiel: Getriebegehäuse

3-stufiges Getriebe



Statische Reduktion auf Lagerstellen



Reduktion auf die 8 Lagerstellen

Vergleich beider Algorithmen bzgl. Rechenzeit & Speicheraufwand:

- (1) Transformationsmatrix-Ansatz
- (2) SCHUR-Komplement-Ansatz

System

- Installierter RAM: 128 GB
- Prozessor: Intel® Core™ i9-10980XE CPU@3.00GHz

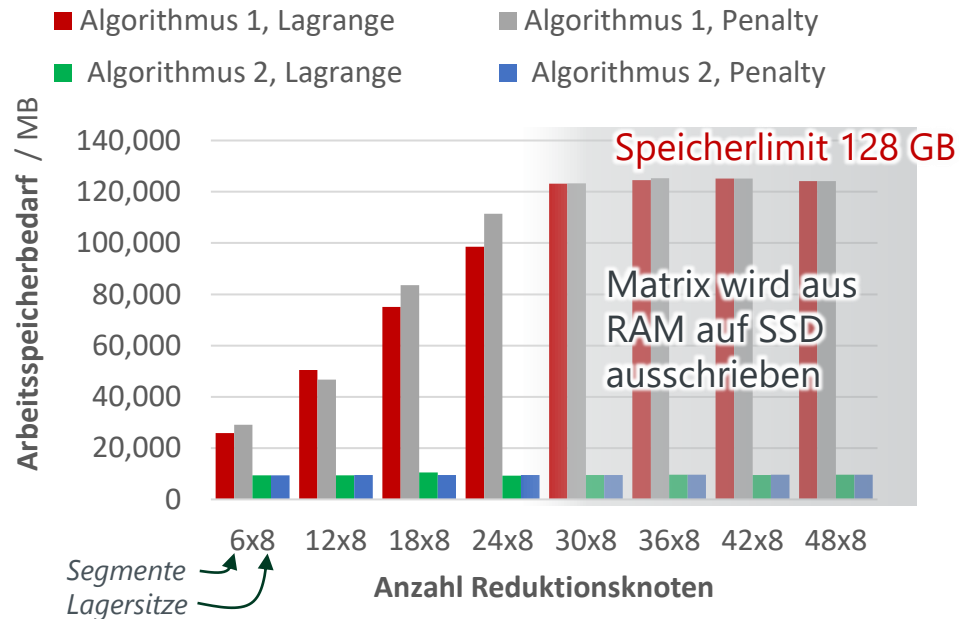
288



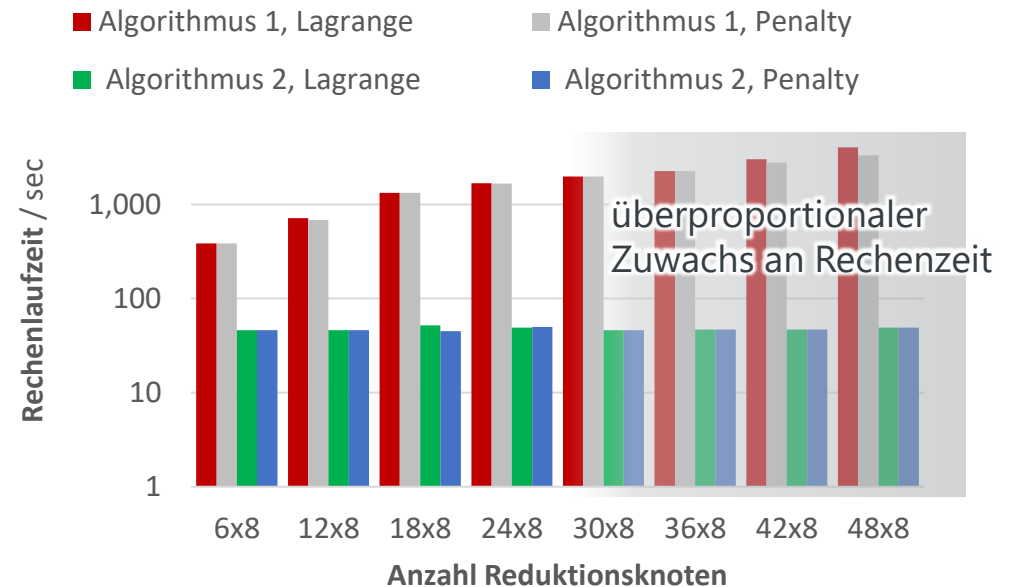
Vergleich der Reduktionsalgorithmen

Anwendungsbeispiel: Getriebegehäuse

Arbeitsspeicherbedarf



Rechenzeit



Z88



Algorithmus 2 deutlich rechenzeit- und speichereffizienter, insbesondere bei großer Anzahl an Master-FG



Zusammenfassung

- **Motivation:** Reduktion von FE-Strukturen für recheneffiziente MKS
- **GUYAN-Reduktion** für quasi-statische Problemstellungen
- Verschiedene **Reduktionsalgorithmen:**
 - (1) Transformationsmatrix-Ansatz
 - (2) SCHUR-Komplement-Ansatz
- Algorithmus 2 deutlich **rechen- & speichereffizienter**, insbesondere bei vielen Master-FG, da Aufbau der Transformationsmatrix umgangen wird

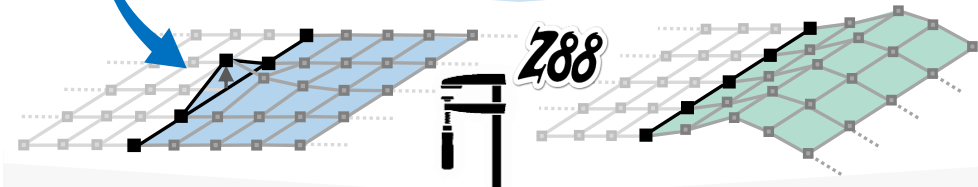
Ausblick: beschleunigte dynamische Reduktion

- **Eigenschwingungsanalyse**

$$(K - \omega^2 M)\Phi_i = 0$$

- **Statische Reduktion**  **beschleunigt**

$$K_{\text{red}} = K_{ee} - K_{ei}K_{ii}^{-1}K_{ie}$$



- **Dynamische Reduktion**

$$u = \begin{bmatrix} I & 0 \\ -K_{ii}^{-1}K_{ie} & \Phi_i \end{bmatrix} \begin{bmatrix} u_e \\ q_k \end{bmatrix} = T_{\text{HCB}}q$$

Freiheitsgrade

i intern
e extern
k modal



Billenstein, D.: Finite-Elemente-Strategien zur Berücksichtigung des elastischen Umfeldes in der Topologieoptimierung – Beitrag zur steifigkeitsoptimierten Auslegung von Antriebstechnikkomponenten auf Basis von Z88. Dissertation. Shaker Verlag, Düren, 2019.

Cook, R.D.; Malkus, D.S.; Plesha, M.E. et al.: Concepts and Applications of Finite Element Analysis. John Wiley & Sons, Hoboken, 2001.

Guyan, R.J.: Reduction of stiffness and mass matrices. In: AIAA Journal 3 (1965), Heft 2, S. 380.

Hüter, F.; Hammerl, G.; Mieth, F. et al.: Erweiterung der statischen Reduktion zur Berücksichtigung von verklebten Kontakten (Statische Reduktion mit verklebtem Bauteilkontakt) – Forschungsvorhaben Nr. 979 I. Abschlussbericht. Lehrstuhl für Konstruktionslehre und CAD, Universität Bayreuth, Frankfurt am Main Ausgabe 2024.

Palazzolo, A. B.: Vibration theory and applications with finite elements. John Wiley & Sons, Inc, Chichester, West Sussex, United Kingdom, 2016.

Qu, Z.-Q.: Model Order Reduction Techniques. Springer London, London, 2004.

Schweigert, D.; Wittmann, J.; Stahl, C. et al.: Erweiterung der Detailtiefe bei der Berücksichtigung von elastischen Gehäusestrukturen in der Getriebeauslegung mit RIKOR (RIKORplusZ88 II) – Forschungsvorhaben Nr. 711 II. Abschlussbericht. FZG TU München; Lehrstuhl für Konstruktionslehre und CAD, Universität Bayreuth, Frankfurt am Main Ausgabe 2021.

Weinberger, U.; Glenk, C.; Stahl, C. et al.: Einbindung elastischer Gehäusestrukturen in die Getriebeauslegung mit RIKOR und Visualisierung des Getriebegesamt-systems in der FVA Workbench (RIKORplusZ88) – Forschungsvorhaben Nr. 711 I. Abschlussbericht. Forschungsstelle für Zahnräder und Getriebesysteme, Technische Universität München; Lehrstuhl für Konstruktionslehre und CAD, Universität Bayreuth, Frankfurt am Main Ausgabe 2017.

Wittmann, J.: Struktur-dynamische Finite-Elemente-Analyse zur Berechnung von Antriebstechnik-Komponenten – Beitrag zur Eigenschwingungsanalyse und Frequenzganganalyse mit Bauteilkontakt, Fortschritte in Konstruktion und Produktion Heft 63, Shaker, Düren, 2024.

Davis, T.A.: Direct methods for sparse linear systems, Fundamentals of algorithms, Society for Industrial and Applied Mathematics, Philadelphia, 2006.

