

STRUCTURAL DESIGN & RESEARCH CENTRE LTD

(Aiming to do all in a click!!)

Dhaka: House - 261 (Echamoti), Flat - 4B
Block - B, Main Road, Bashundhara R/A.
Chattragram: Suite - A9, Ridgedale.
Isphani Gate - 01, Zakir Hussain Road, Khulsi.

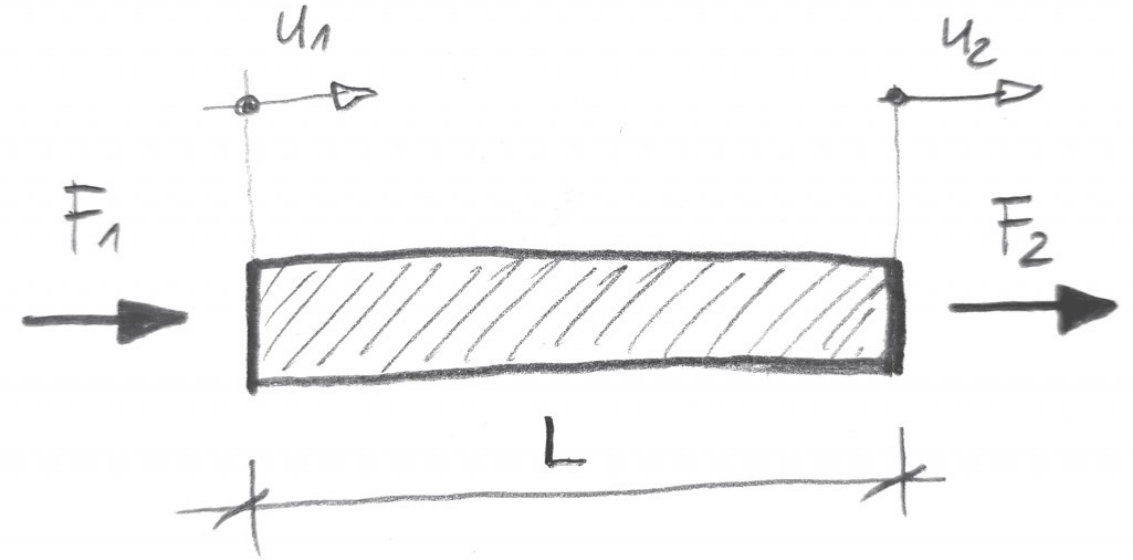
From SDRC Desk:

আপনি হাত দিয়ে FEM করতে পারেন?

আপনি সত্যিই হাতে হাতে FEM করতে পারেন। আসল প্রশ্ন হচ্ছে, আপনি কেন করবেন ?! এমনকি কম্পিউটার সহায়তা দিয়েও আপনি লিনিয়ার বিশ্লেষণের জন্য কমবেশি আবদ্ধ হবেন... কারণ

ননলাইনার স্টাফগুলি বিকাশ করা বেশ কঠিন। তবে কেবল গিগলসের জন্য, কীভাবে এটি করা যায় তা একবার দেখে নেওয়া যাক!
আমি মনে করি প্রতিটি "বেসিক" FEA বই বা গাইড একইভাবে শুরু হয়। ঠিক আছে, তারা "FEA কীভাবে কাজ করে" সম্পর্কে আরও রয়েছে তবে এটি একই জিনিস।

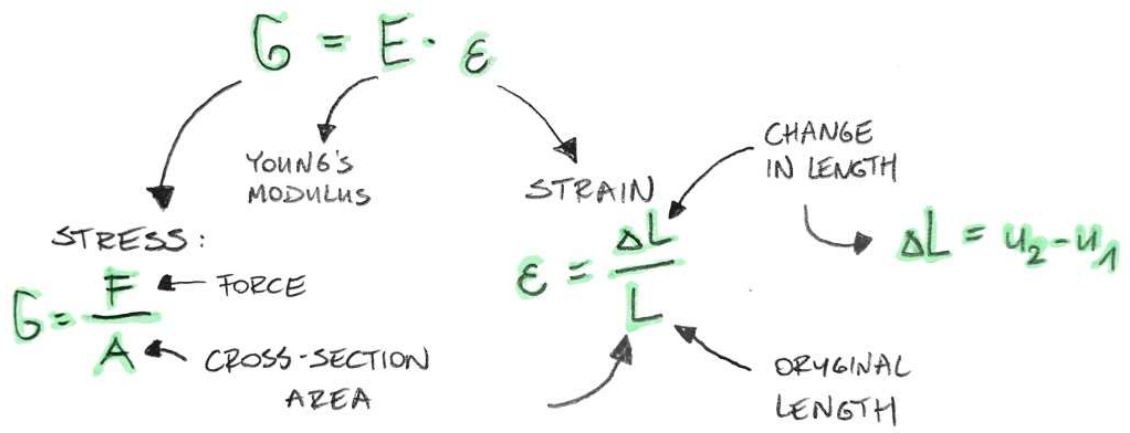
সাধারণত, এই জাতীয় গাইডগুলি রডের উপাদানগুলিতে ফোকাস করে। খুব বেশি মন্তব্য না করেই শুরুটি সাধারণত এমন কিছু হয়:



উপরেরটি হ'ল সম্ভাব্যতম উপাদান: একটি 1 DOF Spring। কী গুরুত্বপূর্ণ তা হল, আমাদের উপাদানটি কেবল টান বা সংকোচন বহন করতে পারে।
প্রথমত, কারণ এই জাতীয় উপাদানগুলি বাঁকানো এবং শিয়ার "দেখবে" না। এটি কেবল একটি সাধারণ Force বহন করছে এবং এর চেয়ে বেশি কিছুই নেই!

এটি সমস্যাটিকে সহজ করে তোলে এবং আমরা সহজেই এমন একটি ক্ষেত্রে গণিতটি অনুসরণ করতে পারি! একবার দেখা যাক!

প্রথমত, কয়েকটি প্রাথমিক সমীকরণ আপনার সহজেই স্বীকৃত হওয়া উচিত। u2-u1= রড প্রসারিত



এখন, আমাদের রডের জন্য এক সেকেন্ডের জন্য ফিরে আসি। আমরা ঠিক এটি ভারসাম্যহীন থাকতে চাই? এর অর্থ হ'ল উভয় প্রান্তে থাকা Force সমান মান হওয়া উচিত, তবে বিপরীত দিকগুলিতে কাজ করা উচিত। যেমন একটি ক্ষেত্রে...

$$F_1 = -\frac{A \cdot E}{L}(u_2 - u_1) = \frac{A \cdot E}{L}(u_1 - u_2)$$

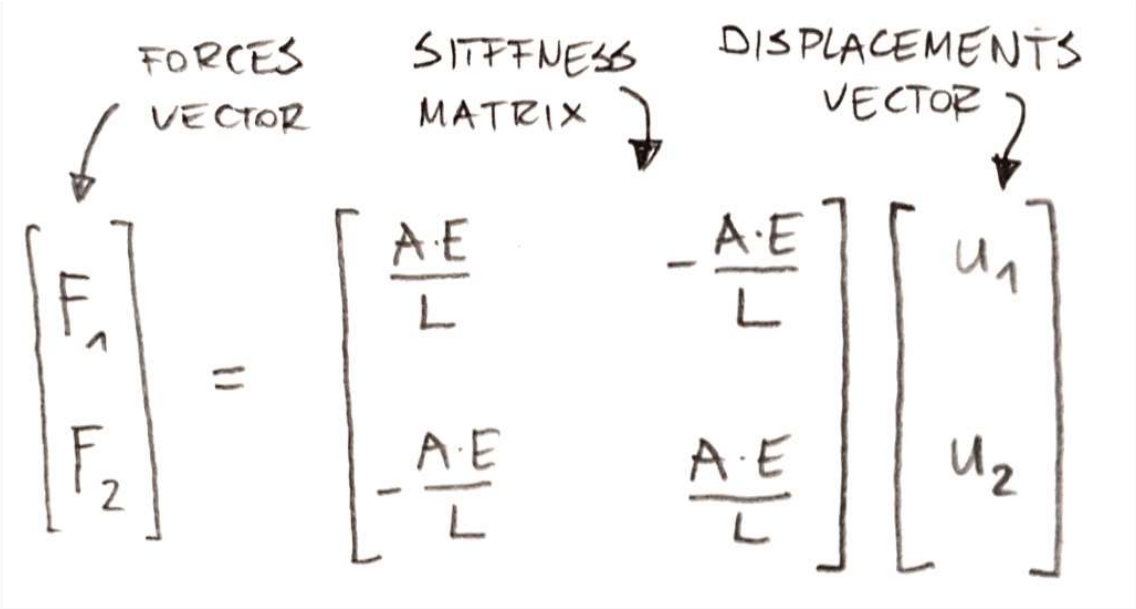
$$F_2 = \frac{A \cdot E}{L}(u_2 - u_1)$$

আপনি খেয়াল করতে পারেন F1 Force "-ve"

ম্যাট্রিক্সস যে জায়গাটিতে জন্মগ্রহণ করেছে
এখনও অবধি আমরা কয়েকটি তুলনামূলকভাবে সুপরিচিত সমীকরণ নিয়ে কিছুটা খেলেছি। এটি আপনার FEA সমাধানকারী যা করেন ঠিক তা নয়, তবে আমরা আরও কাছাকাছি আসছি।

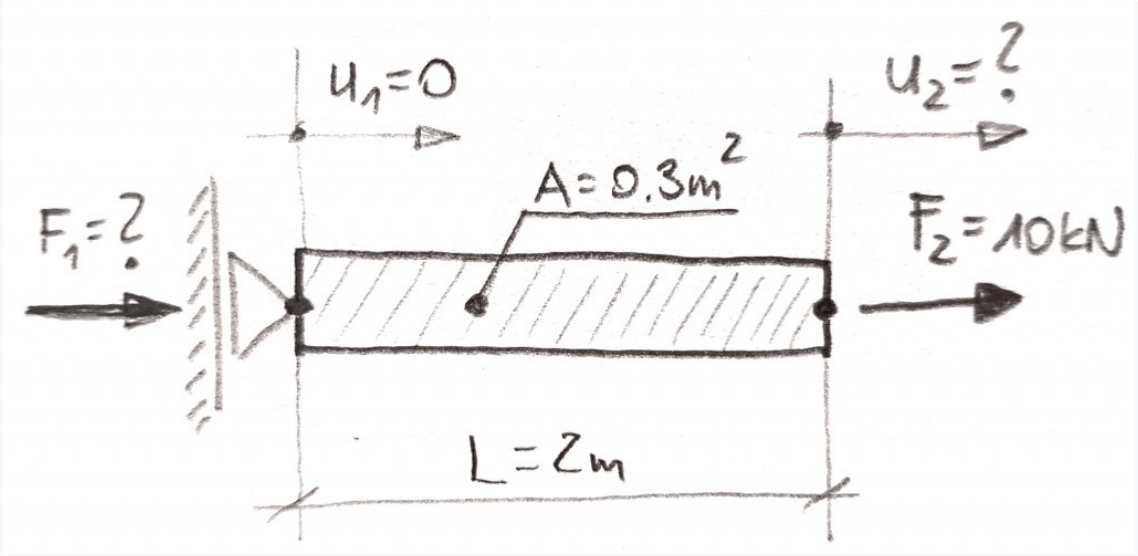
এটি লক্ষ্য করা সহজ, আপনি যদি ফোর্সগুলি (বা বিকৃতি) জানতেন তবে আপনি উপরেরটি তুলনামূলকভাবে সহজে সমাধান করতে পারতেন। এটি কেবলমাত্র 2 "সংযুক্ত" সমীকরণের একটি সেট। তবে যদি তাদের প্রচুর পরিমাণে থাকে (এবং সেখানে থাকবে!) ম্যাট্রিক্স ফর্মের সমাধান করা আরও অনেক সুবিধাজনক। ভাল ... সম্ভবত আপনার জন্য সুবিধাজনক নয়, তবে অবশ্যই আপনার কম্পিউটারে আরও সুবিধাজনক!

উপরোক্ত 2 টি সমীকরণ ম্যাট্রিক্স ফর্মটিতে লেখা খুব সহজ আপনি কেবল Force নিয়ে একটি ভেক্টরে রেখেছিলেন এবং অনুরূপ বিরূপতা দিয়ে আপনি একই কাজ করেন। বাকিটি হ'ল ম্যাট্রিক্সের গুণটি সত্যই:



এখন আপনি জিনিসগুলি লক্ষ্য করা শুরু করতে পারেন। প্রথমত, যদি আমরা বাহকদের মধ্যে Force এবং displacement সংগ্রহ করি তবে কিছু "stuff" বাকী রয়েছে। এই জিনিসগুলি, যেমন আপনি উপরে দেখতে পারেন একটি ম্যাট্রিক্স গঠন করবে। সেখানে থাকা শর্তাদি, মডেলটি কতটা "rigid" তা স্পষ্টভাবে মিলে যায় এবং এ কারণেই এটিকে stiffness ম্যাট্রিক্স বলা হয়। এইভাবে আপনার সমাধানকারী সঠিক উপাদান বৈশিষ্ট্য সহ আপনার মডেলটিতে কোন X-section এবং Thickness ব্যবহার করেছেন তা "জানেন"।

আপাতত, এটি কেবলমাত্র একটি একক উপাদানের (আমাদের ক্ষেত্রে রড) ম্যাট্রিক্স, তবে আমাদের উদাহরণে আরও উপাদান থাকলে আমরা একসাথে একটি "গ্লোবাল" স্ট্রাইনেস ম্যাট্রিক্সে করব।



$$\frac{m^2 \cdot kPa}{m} = \frac{kN}{m}$$

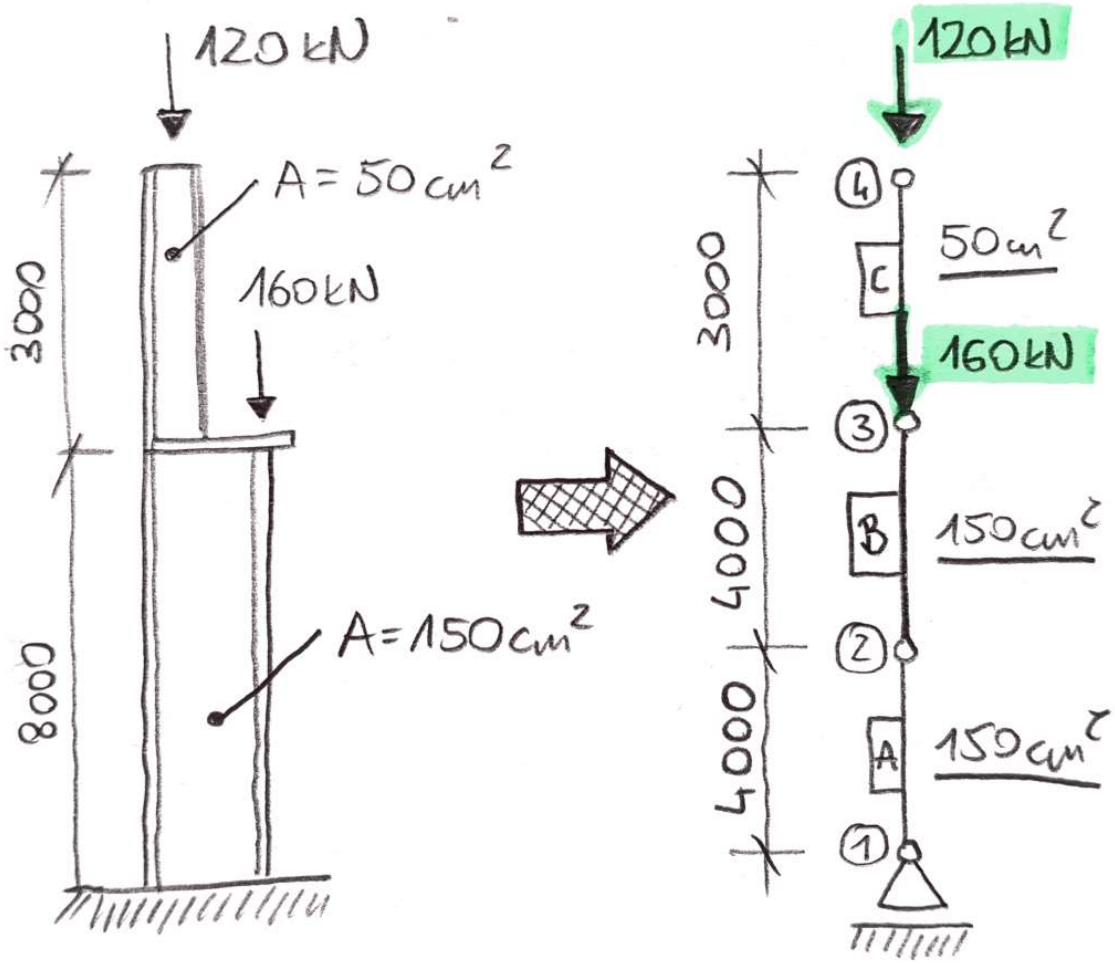
$$\begin{bmatrix} F_1 \\ 10kN \end{bmatrix} = \begin{bmatrix} \frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} & -\frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} \\ -\frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} & \frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} \end{bmatrix} \begin{bmatrix} 0 \\ u_2 \end{bmatrix}$$

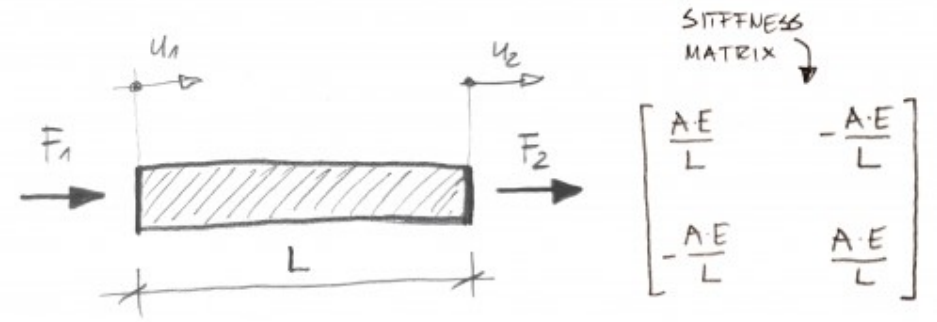
$$\begin{bmatrix} F_1 \\ 10kN \end{bmatrix} = \begin{bmatrix} \frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} & -\frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} \\ -\frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} & \frac{0.3 \cdot 210e^6}{2}\frac{kN}{m} \end{bmatrix} \begin{bmatrix} 0 \\ u_2 \end{bmatrix}$$

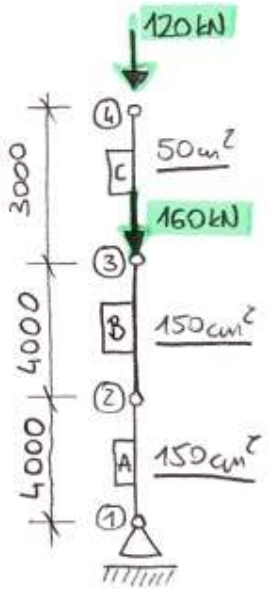
$$10kN = \frac{0.3 \cdot 210e^6}{2} \cdot u_2 \quad \Rightarrow \quad u_2 = 0.000317\,mm$$

$$\Delta L = \frac{F \cdot L}{A \cdot E} = \frac{10kN \cdot 2m}{0.3m^2 \cdot 210e^6\,kPa} = 0.000317\,mm$$

$$F_1 = \frac{0.3 \cdot 210e^6}{2} \cdot 0 - \frac{0,3 \cdot 210e^6}{2} \cdot 0,000317 = -10kN$$







$$\boxed{C}\quad \frac{AE}{L} = \frac{50\,cm^2 \cdot 210e^6\,kPa}{3m} = 350\,\frac{MN}{m} \Rightarrow \begin{bmatrix} 350 & -350 \\ -350 & 350 \end{bmatrix}$$

$$\boxed{B}\quad \frac{AE}{L} = \frac{150\,cm^2 \cdot 210e^6\,kPa}{4m} = 787\,\frac{MN}{m} \Rightarrow \begin{bmatrix} 787 & -787 \\ -787 & 787 \end{bmatrix}$$

$$\boxed{A}\quad \frac{AE}{L} = \frac{150\,cm^2 \cdot 210e^6\,kPa}{4m} = 787\,\frac{MN}{m} \Rightarrow \begin{bmatrix} 787 & -787 \\ -787 & 787 \end{bmatrix}$$

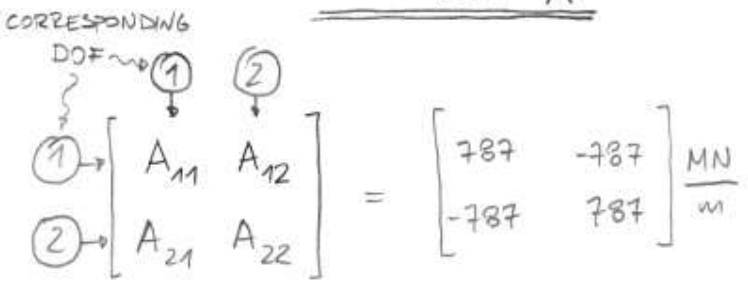
$$\begin{bmatrix} F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} K_{22} & K_{23} \\ K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \end{bmatrix}$$

$$F_2 = K_{22} \cdot u_2 + K_{23} u_3$$

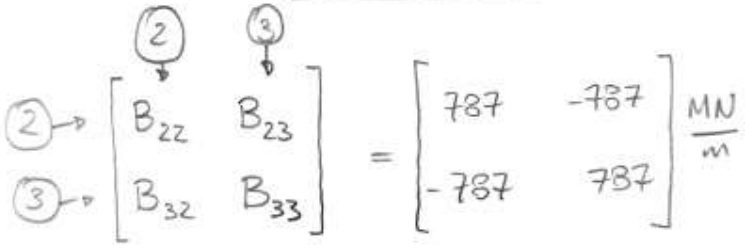
$$F_3 = K_{32} \cdot u_2 + K_{33} \cdot u_3$$

$$\Rightarrow \begin{bmatrix} K_{22} & K_{23} \\ K_{32} & K_{33} \end{bmatrix} \begin{matrix} \leftarrow 2 \\ \leftarrow 3 \end{matrix}$$

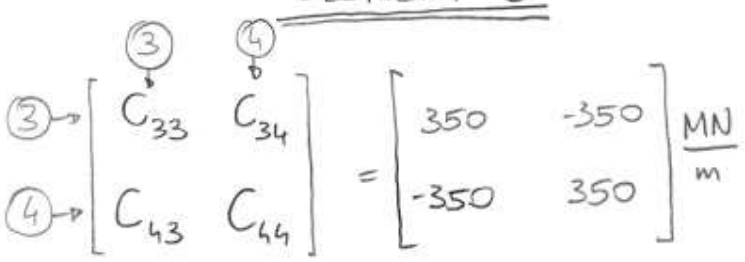
ELEMENT A



ELEMENT B



ELEMENT C



ELEMENT A

CORRESPONDING DOF

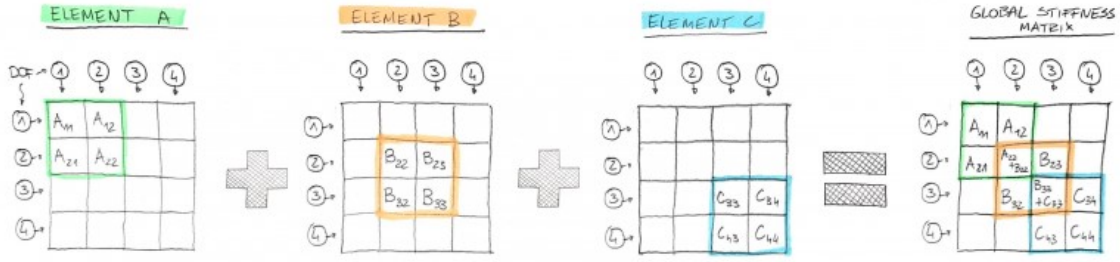
$$\begin{matrix} & 1 & 2 \\ 1 & A_{11} & A_{12} \\ 2 & A_{21} & A_{22} \end{matrix} = \begin{bmatrix} 787 & -787 \\ -787 & 787 \end{bmatrix} \frac{MN}{m}$$

ELEMENT B

$$\begin{matrix} & 2 & 3 \\ 2 & B_{22} & B_{23} \\ 3 & B_{32} & B_{33} \end{matrix} = \begin{bmatrix} 787 & -787 \\ -787 & 787 \end{bmatrix} \frac{MN}{m}$$

ELEMENT C

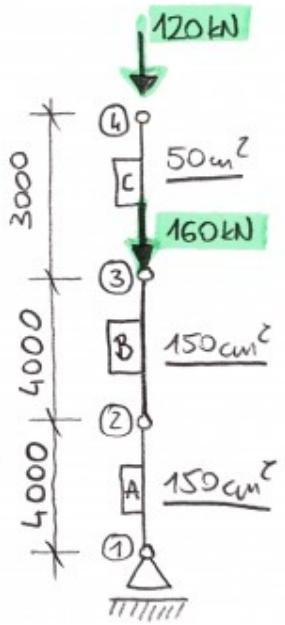
$$\begin{matrix} & 3 & 4 \\ 3 & C_{33} & C_{34} \\ 4 & C_{43} & C_{44} \end{matrix} = \begin{bmatrix} 350 & -350 \\ -350 & 350 \end{bmatrix} \frac{MN}{m}$$



$$A_{22} + B_{22} = 787 + 787 = 1574$$

$$K = \begin{bmatrix} 787 & -787 & 0 & 0 \\ -787 & 1574 & -787 & 0 \\ 0 & -787 & 1137 & -350 \\ 0 & 0 & -350 & 350 \end{bmatrix} \frac{MN}{m}$$

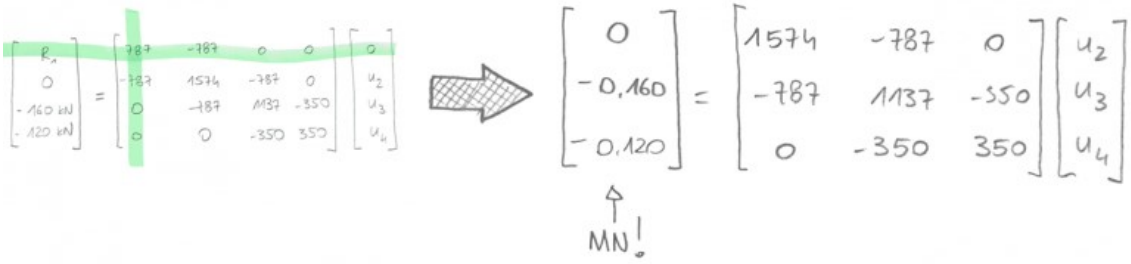
$$B_{33} + C_{33} = 787 + 350 = 1137$$



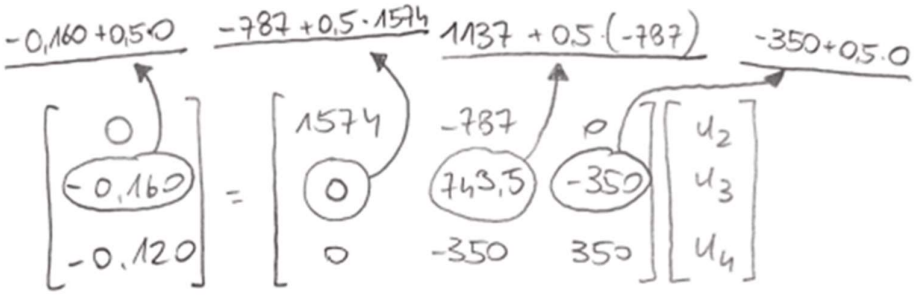
$$F = \begin{bmatrix} R_1 \\ 0 \\ -160\,kN \\ -120\,kN \end{bmatrix} \begin{matrix} \leftarrow 1 \\ \leftarrow 2 \\ \leftarrow 3 \\ \leftarrow 4 \end{matrix}$$

$$u = \begin{bmatrix} 0 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \begin{matrix} \leftarrow 1 \\ \leftarrow 2 \\ \leftarrow 3 \\ \leftarrow 4 \end{matrix}$$

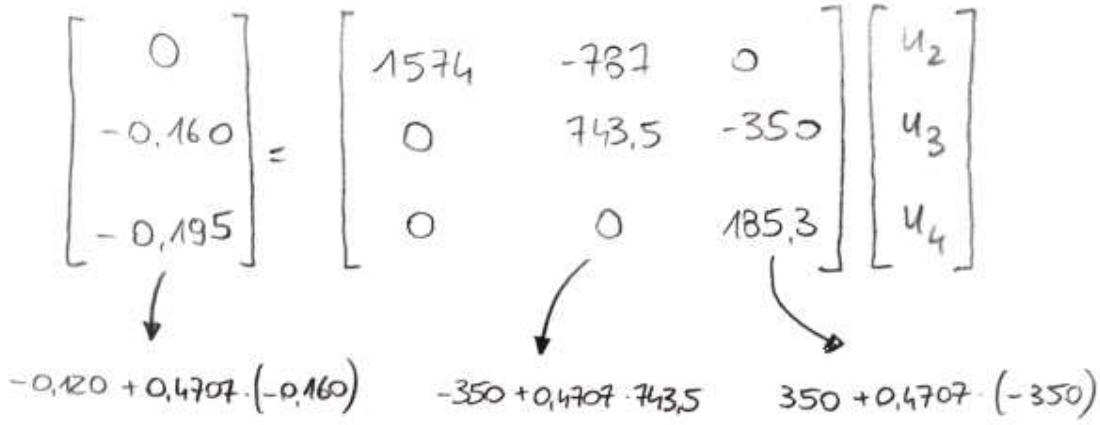
$$\begin{bmatrix} R_1 \\ 0 \\ -160\,kN \\ -120\,kN \end{bmatrix} = \begin{bmatrix} 787 & -787 & 0 & 0 \\ -787 & 1574 & -787 & 0 \\ 0 & -787 & 1137 & -350 \\ 0 & 0 & -350 & 350 \end{bmatrix} \begin{bmatrix} 0 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$$



STEP 1: (Row 2) = (Row 2) + 0.5 (Row 1)



STEP 2: (Row 3) = (Row 3) + 350/743.5 (Row 2) = (Row 3) + 0.4707 (Row 2)



STEP 3: Use what you've got!

$$-0.195\ MN = 185{,}3 \cdot \frac{MN}{m} \cdot u_4 \quad \Rightarrow \quad u_4 = -0{,}00105\ m = -1{,}05\ mm$$

$$-0.160\ MN = 743{,}5\ \frac{MN}{m} \cdot u_3 - 350\ \frac{MN}{m} \cdot (-0{,}00105\ m)$$

$$u_3 = 0{,}00071 = -0{,}71\ mm$$

$$0 = 1574 \frac{MN}{m} \cdot u_2 - 787 \frac{MN}{m} \cdot (-0.00071) \Rightarrow u_2 = -0{,}355\,mm$$

STEP 4: Don't forget about what you crossed out!

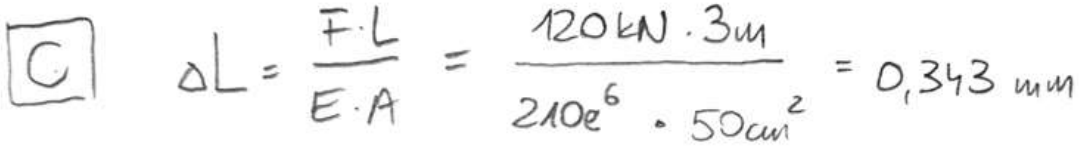
$$R_1 = 787 \frac{MN}{m} \cdot 0 - 787 \frac{MN}{m} \cdot (-0{,}355\,mm) = 279{,}4\,kN$$

মোট load 280kN ছিল, আমি যে reaction পেয়েছি তা 279.4kN। দেখে মনে হচ্ছে আমি ফলাফলের 0.2% মিস করছি। এটি রাউন্ড আপ ত্রুটির কারণে। আমি কেবল 3-4 টি সংখ্যা ব্যবহার করছিলাম এবং সেগুলি অ্যাড হয়ে গেছে। যদিও কোনও উদ্বেগ নয়, কম্পিউটারটি 4 টির বেশি সংখ্যার ব্যবহার করে

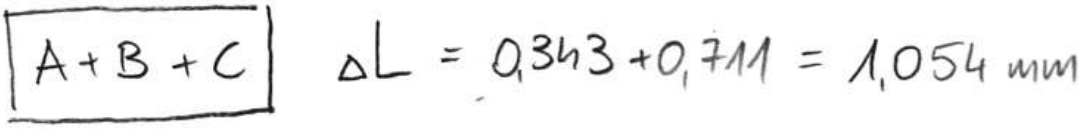
প্রথমত আমি "মূল কলাম" এর সংক্ষিপ্তকরণ গণনা করব। এটি আমাদের ছোট FEA কার্যে এ এবং বি উপাদানগুলির সাথে প্রতিনিধিত্ব করেছিল। এখানে, ধ্রুবক ক্রস-বিভাগ এবং উপাদান সহ তারা ক্রমাগত লোড হওয়ায় এগুলি পৃথকভাবে treat প্রয়োজন নেই। কেবল মনে রাখবেন যে কলামটির এই অংশে মোট প্রয়োগিত লোড শুরুতে অতিরিক্ত 160kN দিয়ে উপরের দিক থেকে 120kN - আমাদের সেগুলি যোগ করতে হবে!

A+B $$\Delta L = \frac{F \cdot L}{E \cdot A} = \frac{280\,kN \cdot 8\,m}{210e^6 \cdot 150\,cm^2} = 0{,}711\,mm$$

total shortening of the “main column” নোডে নিম্নমুখী (যেমন ক্ষেত্রে নেতিবাচক মান) সমান হওয়া উচিত Our আমাদের FEA “solver” spit out −0.71 মিমি – একটি দর্শনীয় সাফল্য!
Shortening of the top “smaller column” alone is:



এটি একা যাচাইয়ের জন্য সহায়ক নয়, তবে কলামটির মোট deformation (নোড 4) উপরের মানগুলির সমষ্টি হতে হবে:

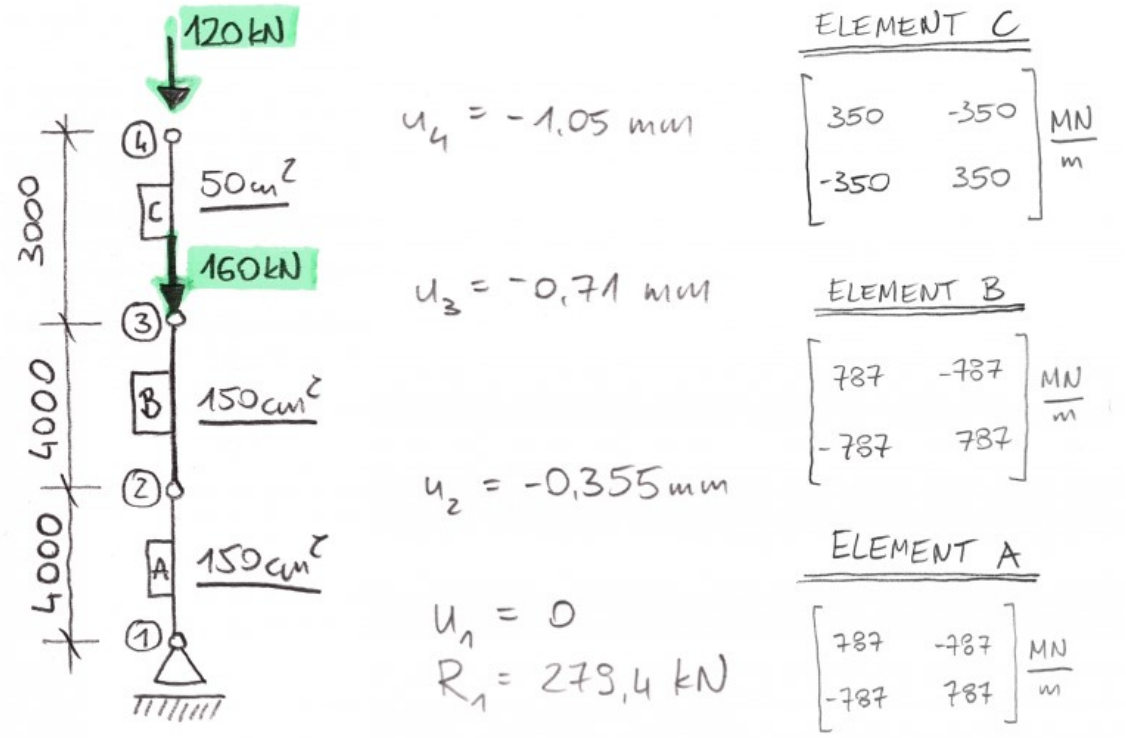


আমাদের FEA SOLVER displacement -1.05 মিমি হিসাবে দেখিয়েছে ... আরও একটি দর্শনীয় জয়!

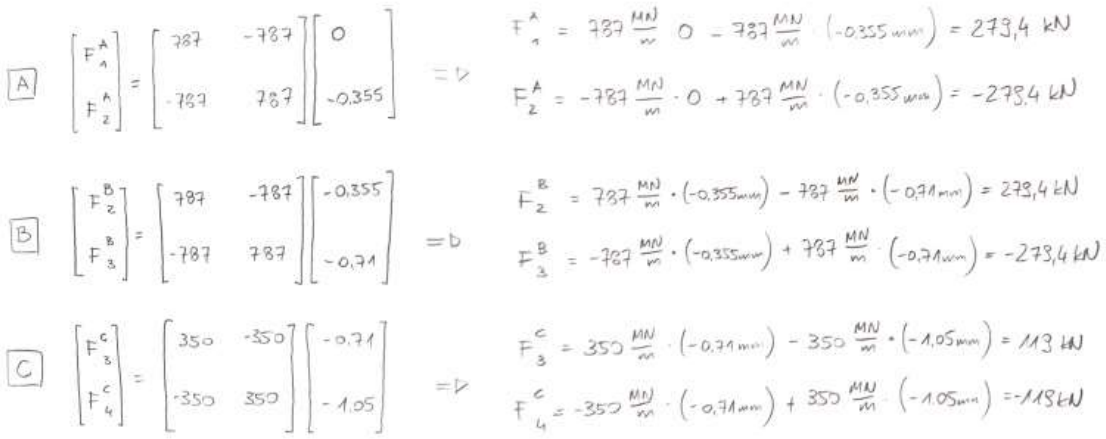
Internal Forces

গ্রেট! উপরে আমরা সিস্টেমের displacements এবং reaction forces সমাধান করতে পরিচালিত করেছি। আমরা ইতিমধ্যে কয়েকটি দরকারী জিনিস জানি। তবে স্পষ্টতই তারা সবাই নয়। এখন, আসুন আশ্চর্য হওয়া যাক যে প্রতিটি উপাদানগুলির **Internal Forces** গণনা করা যায় – সেগুলি ডিজাইনের ক্ষেত্রে কী কাজে আসে?
প্রথমত, আসুন আমরা ইতিমধ্যে যা জেনেছি তার সমষ্টি করি। আমরা আমাদের সিস্টেমে প্রতিক্রিয়া শক্তি এবং স্থানচ্যুতি গণনা করতে পেরেছি এবং প্রক্রিয়াটিতে আমাদের প্রতিটি উপাদানগুলির

জন্য stiffness ম্যাট্রিক্স তৈরি করতে হয়েছিল। ফলাফলগুলির সমস্ত নীচে রয়েছে:



internal forces গণনা আসলে বেশ সহজ। আমরা প্রতিটি পৃথক উপাদানের জন্য equilibrium equation (সামঞ্জস্য সমীকরণ) করব, তবে active forces এর পরিবর্তে, আমরা উপাদানটিতে internal forces ব্যবহার করব। যেহেতু আমরা ইতিমধ্যে deformations জানি, এটি ভালভাবে কাজ করা উচিত। আপনি যখন ভাবছিলেন, আমি যখন সাবস্ক্রিপ্টে একটি অভ্যন্তরীণ শক্তি চিহ্নিত করব তখন একটি নোড নম্বর থাকবে যার কাছাকাছি মানটি "অবস্থিত"। সুপারস্ক্রিপ্টে উপাদান বর্ণ। এইভাবে আমরা এখানে হারিয়ে যাব না:)



আপনি দেখতে পাচ্ছেন, internal forces সন্ধান করা এতটা খারাপ নয়! আমরা আগের মতো কিছু নির্ভুলতা হারাচ্ছি (উপাদান C তে force 120kN এর পরিবর্তে 119kN), তবে এটি কারণ আমি অলস এবং আমি সংখ্যাগুলি দ্রুত গতিতে দেখি!

দ্রষ্টব্য, যে উপাদান C এবং B তে আমরা প্রয়োগকৃত লোডগুলি ব্যবহার করিনি। তবে অভ্যন্তরীণ শক্তির পার্থক্য অবশ্যই রয়েছে। node 3 এলিমেন্ট C 119kN দেখায় তবে এলিমেন্ট B 279.4 দেখায়। পার্থক্যটি হ'ল, node 3 এ (C এবং B উপাদানগুলির মধ্যে "এর মধ্যে মনে করুন) 160kN অতিরিক্ত লোড প্রয়োগ করা হয়!

Strains, Stresses and Shape Functions

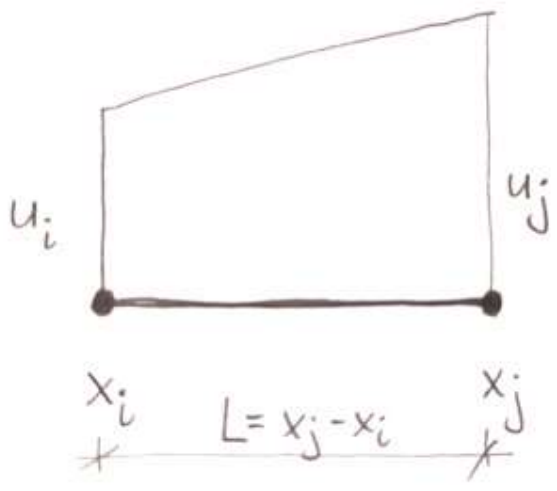
স্ট্রেন (এবং ফলস্বরূপ চাপ) আসলে কোনও নোডাল মান নয়। আপনি অবশ্যই নোডের নিকটবর্তী উপাদানটিতে মান সরবরাহ করতে পারেন (যেমন আমরা internal forces দিয়েছিলাম) তবে একটি twist রয়েছে। "প্রাথমিক দৈর্ঘ্য" সহ কিছু উপস্থিত হওয়ার জন্য স্ট্রেনের দৈর্ঘ্য অবশ্যই পরিবর্তন করতে হবে।

নোডের দৈর্ঘ্য মোটেও নেই ... তাই স্ট্রেনগুলি নোডের পরিবর্তে উপাদানগুলির সাথে যুক্ত।

The twist ... আমরা এখনও জানি না যে উপাদানগুলি কীভাবে deform (বিকশিত হয়!) আমি বোঝাতে চাইছি যে আমরা প্রতিটি উপাদানটির শুরু এবং শেষকে কীভাবে বিকশিত করে তা আমরা জানি এবং আমরা রৈখিক উপাদান ব্যবহার করছি, সুতরাং এটি বেদনাদায়ক স্পষ্ট। তবে আমাদের সমাধানকারী জানেন না যে উপাদানগুলি কীভাবে বিকশিত হয় ... এবং যদি আমরা এটি ব্যবহার করতে চাই তবে আমাদের এ সম্পর্কে কিছু করা দরকার!

আমাদের মূল্যবান মেশিনটিকে সহায়তা করতে আমাদের প্রতিটি উপাদানটিতে শুরুতে এবং শেষে কী মান রয়েছে তার ভিত্তিতে কী ঘটছে তা "ব্যাখ্যা" করতে হবে। এই কি Shape (আকৃতি) ফাংশন জন্য হয়!

যেহেতু আমরা একটি লিনিয়ার উপাদান ব্যবহার করছি, আকৃতির ফাংশনগুলি তুলনামূলক সহজ। একটি সাধারণ উপাদানটি দেখতে কেমন:



আমি “i” টিকে" শুরুর সংখ্যা "হিসাবে এবং“j” শেষ সংখ্যা "হিসাবে ব্যবহার করছি। স্পষ্টতই উপাদানটির উপর নির্ভর করে বিভিন্ন DOF শুরু এবং শেষের দিকে থাকবে সুতরাং "আসল" নম্বর ব্যবহার করা সেরা পছন্দ নয়!
এটি খুব সাধারণ রৈখিক র‍্যাম্প, তবে আমরা মিশ্রণে স্থানাঙ্ক ব্যবহার করতে চাই, সুতরাং সমাধানকারী জানেন যে এটি কোথায়। এছাড়াও, একটি "সমীকরণ" পরিবর্তে, আমি একটি ভেক্টর প্রস্তুত করব, আপনার এই deformation (বিকৃতি) ভেক্টরের সাথে ব্যবহার করা উচিত। এইভাবে এটিতে ইতিমধ্যে আমাদের যে বিকৃতি ভেক্টর রয়েছে তা ব্যবহার করার বিকল্প থাকবে! এই ভেক্টরটি আমাদের উপাদানটির জন্য আকৃতি ফাংশন। সাধারণত, এটি “N” হিসাবে বর্ণনা করা হয়:

$$[N(x)] = \left[\frac{x_j - x}{x_j - x_i} \quad \frac{x - x_i}{x_j - x_i}\right]$$

IF $x_i = 0$ THEN:

$$[N(x)] = \left[1 - \frac{x}{L} \quad \frac{x}{L}\right]$$

দ্রষ্টব্য, x = xi এর জন্য প্রথম মানটি 1, যখন দ্বিতীয়টি 0 হয় x একই নোটে x = xj এর জন্য প্রথম মান 0 হয়, এবং দ্বিতীয়টি 1 হয় কারণ আকৃতি ফাংশন সর্বদা "ক্যালিব্রেটেড" থাকে এইভাবে, কোনও নোডে, কেবল সেই নোডের মানই ফলাফলকে প্রভাবিত করে। এটি আরও দেখা যায় যে x এর মান নির্বিশেষে, ভেক্টরের উভয় পদই ১ টি যোগ করবে আকারের ফাংশনগুলি সম্পর্কে চিন্তা করুন, কোনও শারীরিক মান হিসাবে নয় ... এটি কেবল

"ছড়িয়ে দেওয়া এবং গড়" নোডাল মানগুলির একটি গাণিতিক সরঞ্জাম সম্পূর্ণ সীমাবদ্ধ উপাদান।
অবশ্যই, তার নিজের আকারে ফাংশন অকেজো। তবে এটি ব্যবহার করে আমরা পুরো উপাদানটির বিকৃতি অবস্থা বর্ণনা করতে পারি। এই জাতীয় বিকৃতি স্থানাঙ্ক X এর ফাংশন হবে কারণ এক্স মানের উপর নির্ভর করে উপাদানটির উপর প্রদত্ত বিন্দুতে বিকৃতিটি পৃথক হবে। আসুন "elemental" deformations U(x), কল করুন, যখন নোডাল deformation নিম্নের ক্ষেত্রে u কে আগের মতো ব্যবহার করবে। যেমন একটি ক্ষেত্রে:

$$[U(x)] = [N(x)] \cdot [u]$$

উপরেরটি দেখতে সুন্দর দেখাচ্ছে, তবে এর যদি কিছু ভাল হয় তবে দ্রুত পরীক্ষা করি। উপাদান "B" ভাল প্রার্থী বলে মনে হচ্ছে। এটি নোড 2 (u2 = −0.355 মিমি) দিয়ে শুরু হয় এবং নোড 3 (u3 = −0.71 মিমি) দিয়ে শেষ হয়। এটি একেবারেই সুস্পষ্ট যে উপাদানটির মাঝখানে deformation টি −0.533 মিমি হওয়া উচিত, ভাল, আসুন আমরা আমাদের সমাধানকারীকে এটি সুস্পষ্ট কিনা

$$[U(x)] = \begin{bmatrix} \frac{x_j - x}{x_j - x_i} & \frac{x - x_i}{x_j - x_i} \end{bmatrix} \begin{bmatrix} u_2 = -0.355 \\ u_3 = -0.71 \end{bmatrix}$$

অবশ্যই, আমরা ইতিমধ্যে স্থানাঙ্কগুলি জানি। আমি এখানে global ব্যবহার করব, ধরে নিলাম যে কলামের নীচে x = 0 রয়েছে। এর অর্থ Xi, এলিমেন্ট B এর শুরুতে হবে xi = 4m, xj = 8m এবং আমরা যে জায়গাটি চাই (উপাদানটির মাঝামাঝি) x = 6 মি। এটি মাথায় রেখে আমি কেবল উপরের সমাধান করতে পারি:

$$[u(x)] = \begin{bmatrix} \frac{8-6}{8-4} & \frac{6-4}{8-4} \end{bmatrix} \begin{bmatrix} -0,355 \\ -0,71 \end{bmatrix} = \begin{bmatrix} 0,5 & 0,5 \end{bmatrix} \begin{bmatrix} -0,355 \\ -0,71 \end{bmatrix} =$$

$$= 0,5 \cdot (-0,355) + 0,5 \cdot (-0,71) = -0,533\,mm$$

দেখে মনে হচ্ছে আমাদের সমাধানকারীটির একটি নতুন কৌশল আছে – এটি সীমাবদ্ধ উপাদানগুলির সাথে কীভাবে স্থানচ্যুতিগুলি বিতরণ করতে হয় তা জানে। নিজে থেকেই, এটি সমস্ত অভিনব নয়, তবে এখানে একটি ঝরঝরে জিনিস চলছে। আর তা হ'ল স্ট্রেন হ'ল বাস্তুচ্যুত হওয়ার উপকরণ! এখন, যেহেতু আমাদের displacement function [স্থানচ্যুতি ফাংশন] রয়েছে, আমরা আসলে গাণিতিকভাবে স্ট্রেন গণনা করতে পারি ... অবশেষে আমরা কোথাও পাচ্ছি! সাধারণত একটি ম্যাট্রিক্স [D] থাকে যা বিবরণ দেয় derivatives গুলি ব্যবহার করে কোনটি displacements তে (সত্যিকারের 3D সমস্যায়)। তবে এখানে, আমরা জানি যে আমরা একটি d/dx করব, যেহেতু এটিই আমাদের একমাত্র স্থানাঙ্ক! নির্বিশেষে, ফলাফলটি সাধারণত [B] দিয়ে বোঝানো হয়, সুতরাং আসুন classical notation (শাস্ত্রীয় স্বরলিপিটি) রাখুন:

$$\varepsilon = \frac{d}{dx}[U(x)] = \frac{d}{dx}[N(x)][u] = [B(x)][u]$$

$$[B(x)] = \frac{d}{dx}[N(x)] = \left[\frac{-1}{x_j - x_i} \quad \frac{1}{x_j - x_i}\right] = \left[\frac{-1}{L} \quad \frac{1}{L}\right]$$

উপরে আমি [u] কে derivative থেকে "ছুঁড়ে" ফেলে দিতে পারি, কারণ এগুলি constant values of nodal deformations। লক্ষ করুন, B (x) আমাদের ক্ষেত্রে x উপর নির্ভর করে না! আমরা linear distribution of deformation along the length সহ উপাদানগুলি ব্যবহার করছি ... ফলে ফলস্বরূপ স্ট্রেনগুলি পুরো উপাদানটির উপর স্থির থাকবে!

আসুন দেখুন আমরা কী স্ট্রেন পাব:

$$\varepsilon = [B(x)][u] = \left[\frac{-1}{L} \quad \frac{1}{L}\right] \begin{bmatrix} -0.355 \\ -0.71 \end{bmatrix} =$$

$$= -\frac{-0.355mm}{4m} + \frac{-0.71mm}{4m} = -0.008875\,\%$$

এটি দেখতে সহজ, শেষ অবধি, সমস্ত ক্রিয়াকলাপ সঠিকটির দিকে পরিচালিত করে : displacement at the end minus the displacement at the beginning divided by length । দেখে মনে হচ্ছে আমরা স্ট্রেনগুলিও গণনা করতে পারি! একই পদ্ধতি অনুসরণ করে, উপাদান A তে স্ট্রেন একই এবং B

উপাদানটিতে একই এবং উপাদান C তে স্ট্রেন হয় 0.011333%।

হৃদয় নিন!

কেবল stresses অনুপস্থিত এবং আমরা একে একদিন ডাকব! ভাগ্যক্রমে আমাদের জন্য, স্ট্রেস এবং স্ট্রেনের মধ্যকার সম্পর্কটি সর্বোপরি আদিম। অবশ্যই, FEA সলভারের "সম্ভাব্য" উপাদানগুলির পরামিতিগুলির একটি ম্যাট্রিক্স থাকা উচিত তবে আমাদের ক্ষেত্রে এটি স্পষ্ট যে আমরা Young's modulus টি ব্যবহার করব:

$$\sigma = E \cdot \varepsilon$$

A $\quad \sigma = 210\,GPa \cdot (-0{,}011333\%) = -23{,}8\,MPa \approx \frac{-120\,kN}{50\,cm^2} = -24\,MPa$

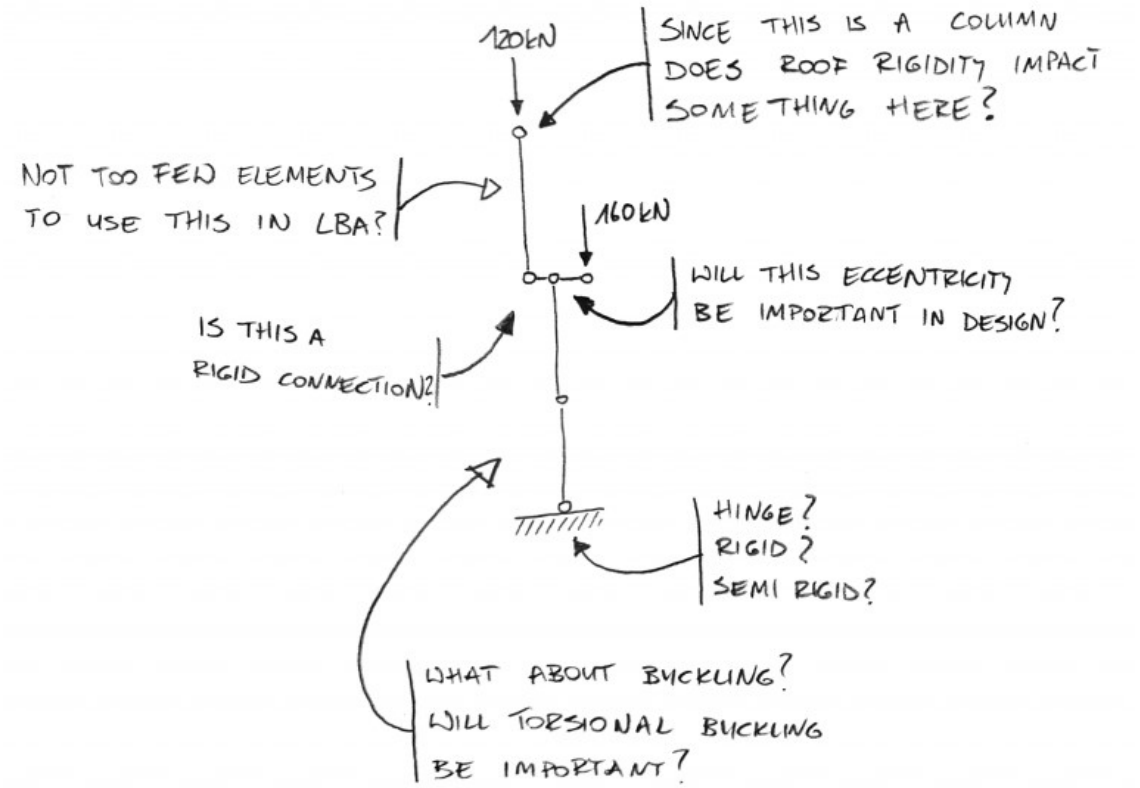
B & C $\quad \sigma = 210\,GPa \cdot (-0{,}008875\%) = -18{,}6\,MPa \approx \frac{-280\,kN}{150\,cm^2} = -18{,}67\,MPa$

অবশেষে, আমি এখানে যা করতে চাইতাম সবই করেছি... এবং এটিই ছিল কিছু! উদযাপন করতে এক মিনিট সময় নিন, তবে আপনি যা জানতে চান তা আমি এখানে সবচেয়ে গুরুত্বপূর্ণ বলে মনে করি... পড়ুন!

একটি নিষ্ঠুর টুইস্ট!

আমি অবশ্যই স্বীকার করতে পারি যে আমি অবাক হয়েছি যে আমি এটিকে বন্ধ করে দিয়েছি। আমি কখনই হাতে হাতে FEA করি নি, এবং এই কাজটি করতে বইগুলিতে সামগ্রী খুঁজে পেতে আমার বেশ কয়েক ঘন্টা লেগেছিল। তবে আমি এটি করেছি এবং আমি অবশ্যই নতুন কিছু শিখেছি। আমি মনে করি যে

এই পোস্টে আমি গত 3 বছরে লিখিত অন্যান্য পোস্টগুলির চেয়ে আরও সমীকরণ রয়েছে
তবে আমার হয়ে, আমি যখন গণিতে আসি তখন একটি এতিহ্যবাহী উপায়ে শেষ করতে চাই: এটি কি দরকারী?
উত্তরটি: না, সত্যই নয়। আমি বোঝাতে চাইছি যদি আমি নিজের সমাধানকারী লেখার লক্ষ্য রাখি তবে সম্ভবত এটিই প্রথম পদক্ষেপ। তবে আমি যদি সেই কলামটির উদাহরণটি ডিজাইন করতে চাই? হাস্যকর হয়ে উঠবেন না - এটি এমনভাবে হয় না! এমনকি যদি আমি উদাহরণটি 6 ডিএফ-তে প্রসারিত করি তবে অ্যাকাউন্টটি বাঁকতে নিতে এটি "নিখুঁত" থেকে দূরে থাকবে।
ডিজাইনের দৃষ্টিকোণ থেকে এই সমস্যাটি সমাধানের জন্য এখানে কয়েকটি প্রশ্ন গুরুত্বপূর্ণ ... যা আমি আজ লিখেছি তার সাথে কোনও সম্পর্ক নেই। এগুলির উত্তর দেওয়া ম্যাট্রিক্সগুলিতে কীভাবে পরিচালনা করা যায় তা শেখার চেয়ে অনেক বেশি গুরুত্বপূর্ণ। সর্বোপরি, আপনি অন্য কারও দ্বারা সম্পন্ন Solver টি ব্যবহার করতে পারেন (প্রচুর পরিমাণে এমনকি Open Source রয়েছে), তবে কোনও একক সমাধানকারী আপনাকে এই প্রশ্নের উত্তর দেবে না:



দেখুন, আমি স্বীকার করি যে এটি অনেকটা দূরে। আমি জানি, উপরেরগুলি আজকের বিষয়টিকে মোটেও কভার করে না, এবং পরিবর্তে FEA র চেয়ে "Steel Design" যেতে পারে। বিশ্বাস করুন, আমি সত্যই জানি যে এই জাতীয় নকশাগুলি অনুসরণ করার জন্য কোড এবং নিয়ম রয়েছে এবং একটি নির্দিষ্ট সফ্টওয়্যার রয়েছে যা আপনাকে এইরকম কোনও কাজে কিছুটা ক্ষেত্রে সহায়তা করবে।

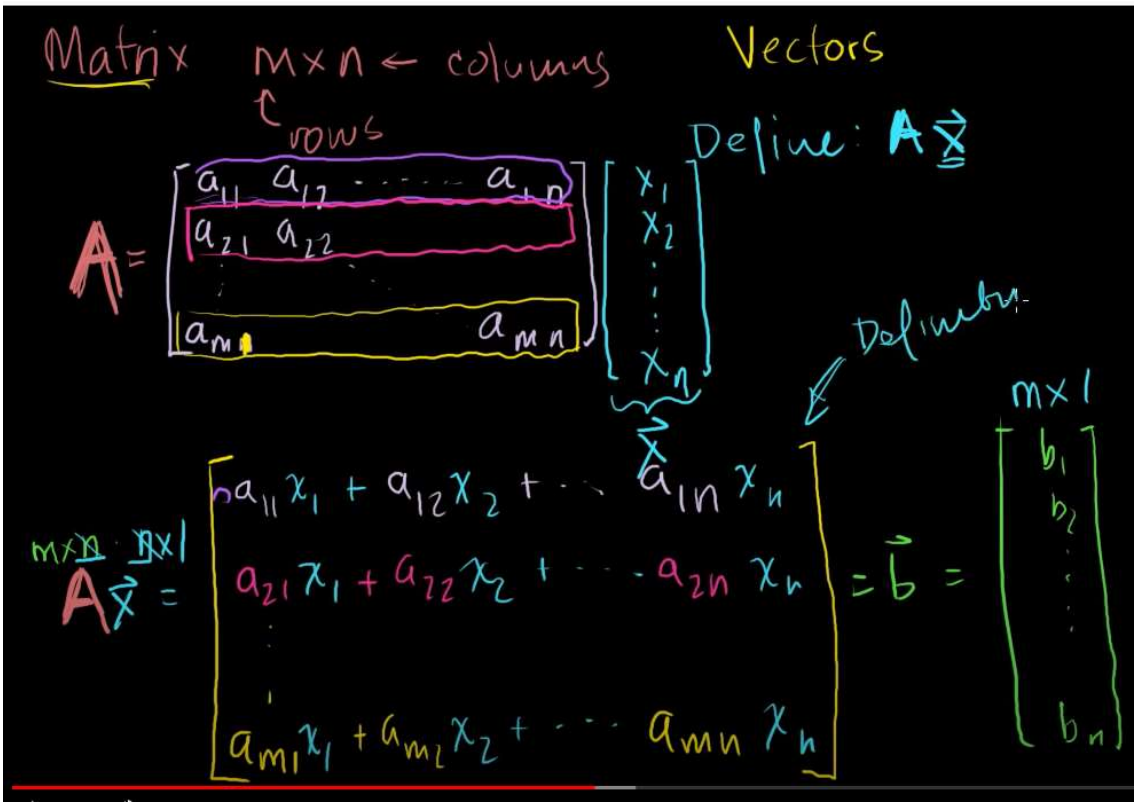
আমি এটি এখানে লিখেছি কারণ আমি দেখতে পেয়েছি যে আরও বেশি লোকেরা FEA করেন তবে কী চলছে সে সম্পর্কে কোনও ধারণা নেই। আমি বোঝাতে চাইছি আমি সবাই "FEA গণতন্ত্রকরণ" এর পক্ষে। আমার জন্য, FEA একটি দুর্দান্ত সরঞ্জাম যা আমার ক্যারিয়ারকে ব্যাপকভাবে সহায়তা করেছিল। আমি চাই সবাই যদি আরও সফল হয় তবে আমি আছি।

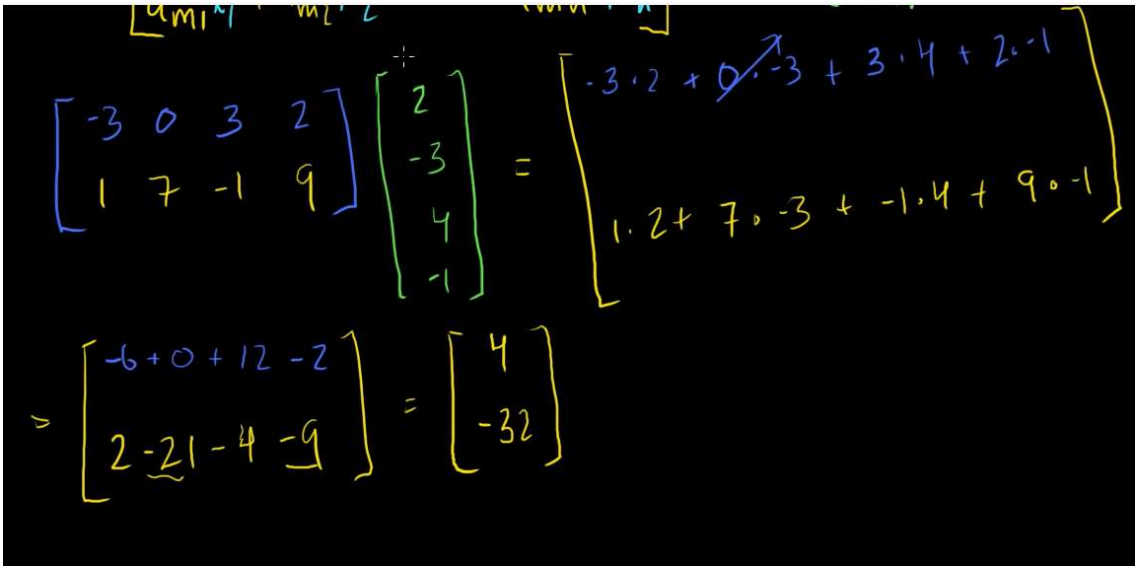
কিন্তু... এটি এখনও জোরে কান্নার জন্য ইঞ্জিনিয়ারিং! আপনি কীভাবে ম্যাট্রিক্সকে চারপাশে স্থানান্তর করতে জানেন তা বোঝানোর অর্থ এই নয় যে আপনি কীভাবে Proper ডিজাইন করবেন জানেন! আসলে, আমি বলব যে সেগুলি দুটি সম্পূর্ণ ভিন্ন জিনিস!

যদি আপনি কেবল শুরু করছেন, দয়া করে (দয়া করে!) মনে রাখবেন – FEA একটি ক্যালকুলেটরের মতো, এটি কেবল গণিতের কাজ করে। ডিজাইনিং এর চেয়ে অনেক জটিল এবং গণিতের ক্রিয়াকলাপগুলির বাইরেও প্রচুর জ্ঞান প্রয়োজন! আপনার মডেলটি কীভাবে জাল করবেন, কীভাবে লোড করবেন এবং কীভাবে এটি সমর্থন করবেন তা আপনাকে বুঝতে হবে। শেষ অবধি, আপনার বিশ্লেষণটি কী "সঠিক এক" এবং ফলাফলগুলি কীভাবে ব্যাখ্যা করতে হয় (এবং কী মনোযোগ দিতে হবে) তা জানতে হবে। ডিজাইনের ক্ষেত্রে কেবলমাত্র অনেকগুলি বিষয় রয়েছে, যা আমি আজ ব্যাখ্যা দিয়েছি তার সাথে কিছুই করার নেই।

আমি জানি যে University তে তারা বেশিরভাগই (কেবল না তবে) এই পোস্টটিতে এই পদ্ধতিতে এমনভাবে কথা বলে... তবে এর অর্থ এই নয় যে এটিই কেবল বিদ্যমান। এ থেকে দূরে ... এটি এমনকি শুরু নয়। সর্বোপরি, আমি এখন এক দশক ধরে সাফল্যের সাথে ডিজাইনে FEA ব্যবহার করছি... এবং কয়েক ঘন্টা আগে কীভাবে এই ফ্রিকিং সমীকরণগুলি সমাধান করতে হয় তা শিখেছি...

"ম্যাট্রিক্সে গণনা করার কিছু টিপস"





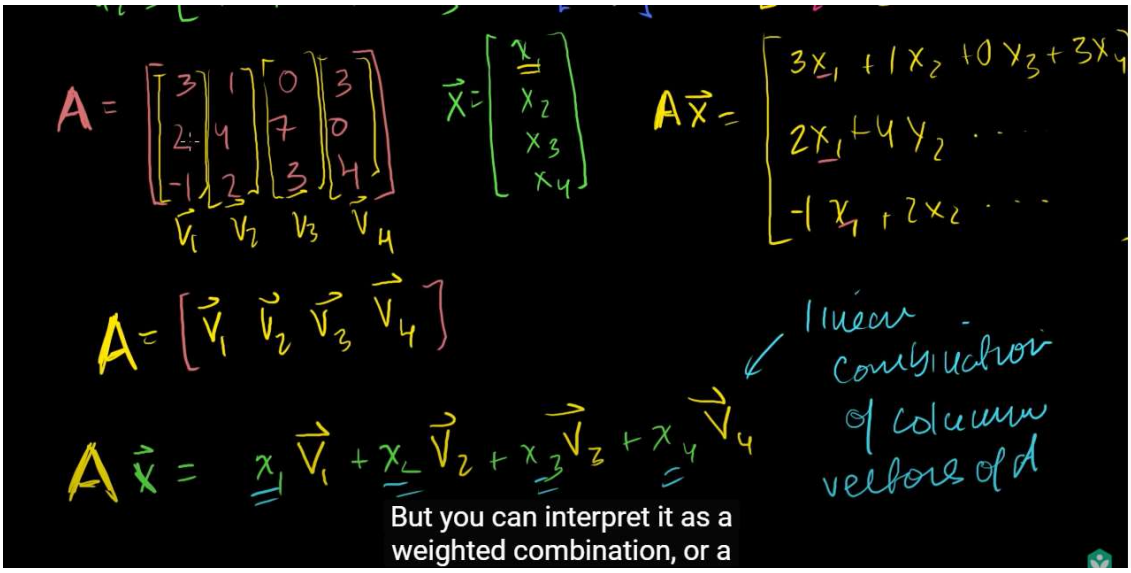
$$\vec{a}_1 = \begin{bmatrix} -3 \\ 0 \\ 3 \\ 2 \end{bmatrix} \quad \vec{a}_2 = \begin{bmatrix} 1 \\ 7 \\ -1 \\ 9 \end{bmatrix}$$

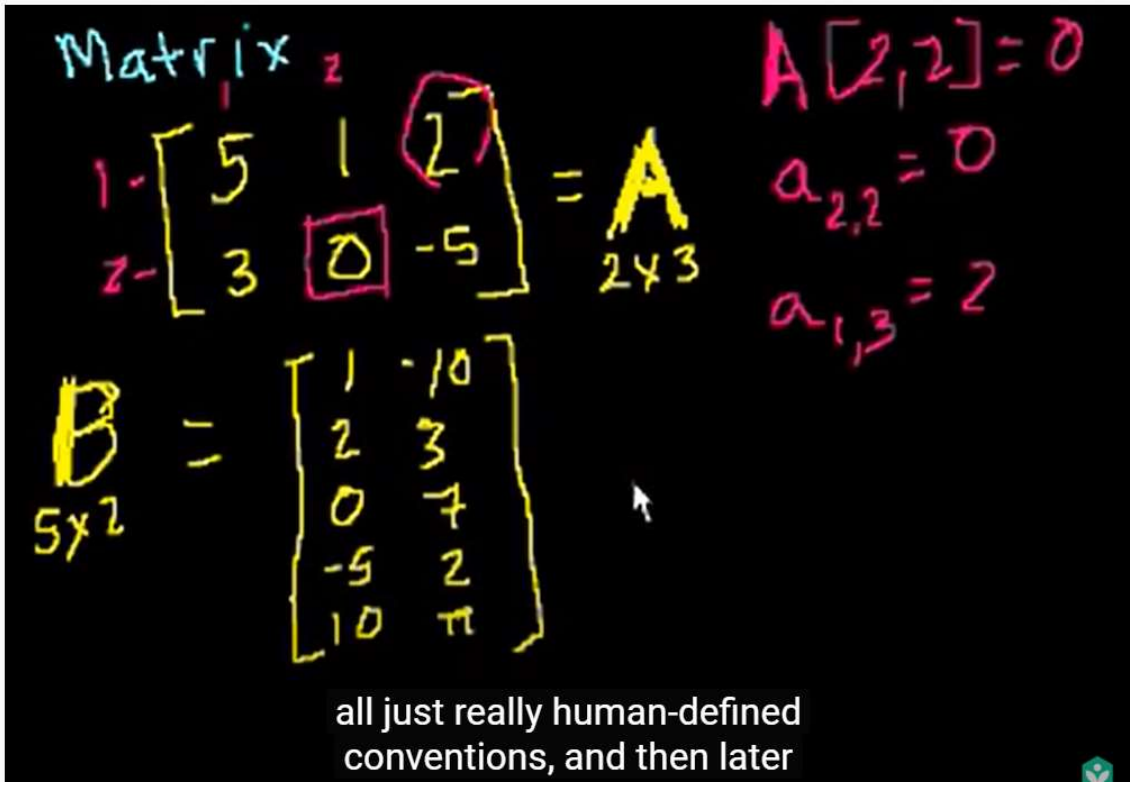
$$a_1^T = [-3 \quad 0 \quad 3 \quad 2]$$

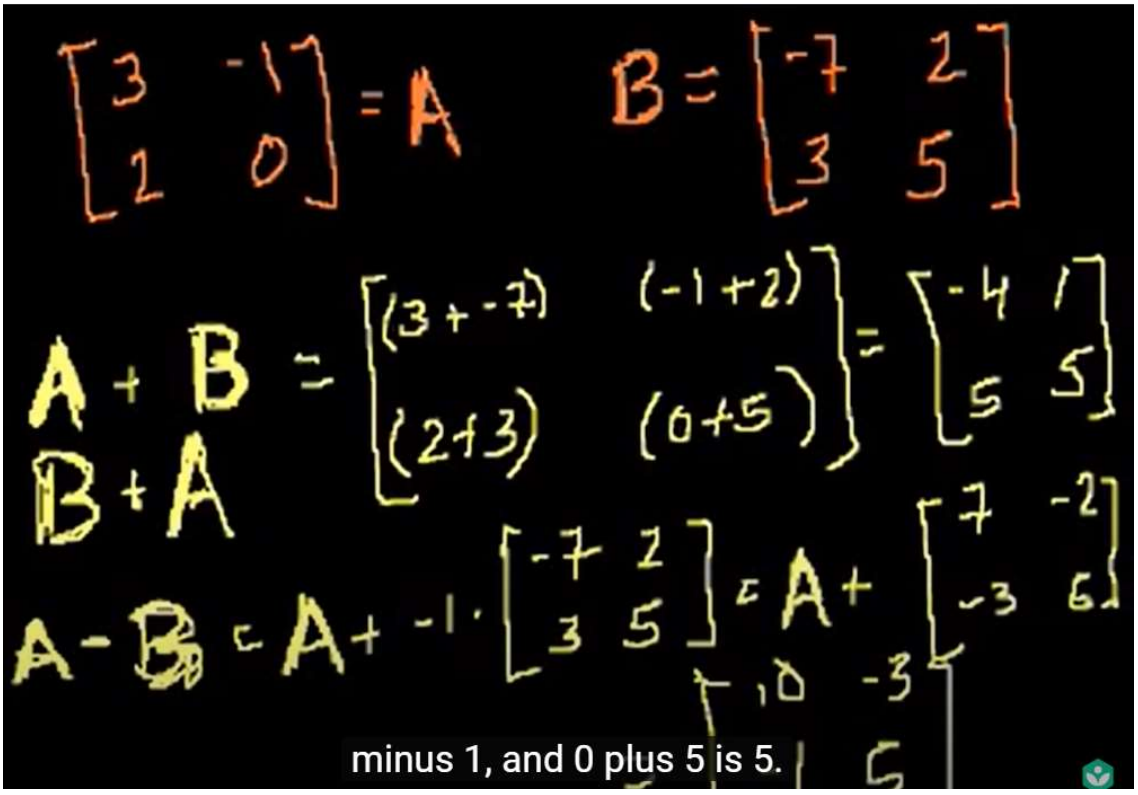
$$a_2^T = [1 \quad 7 \quad -1 \quad 9]$$

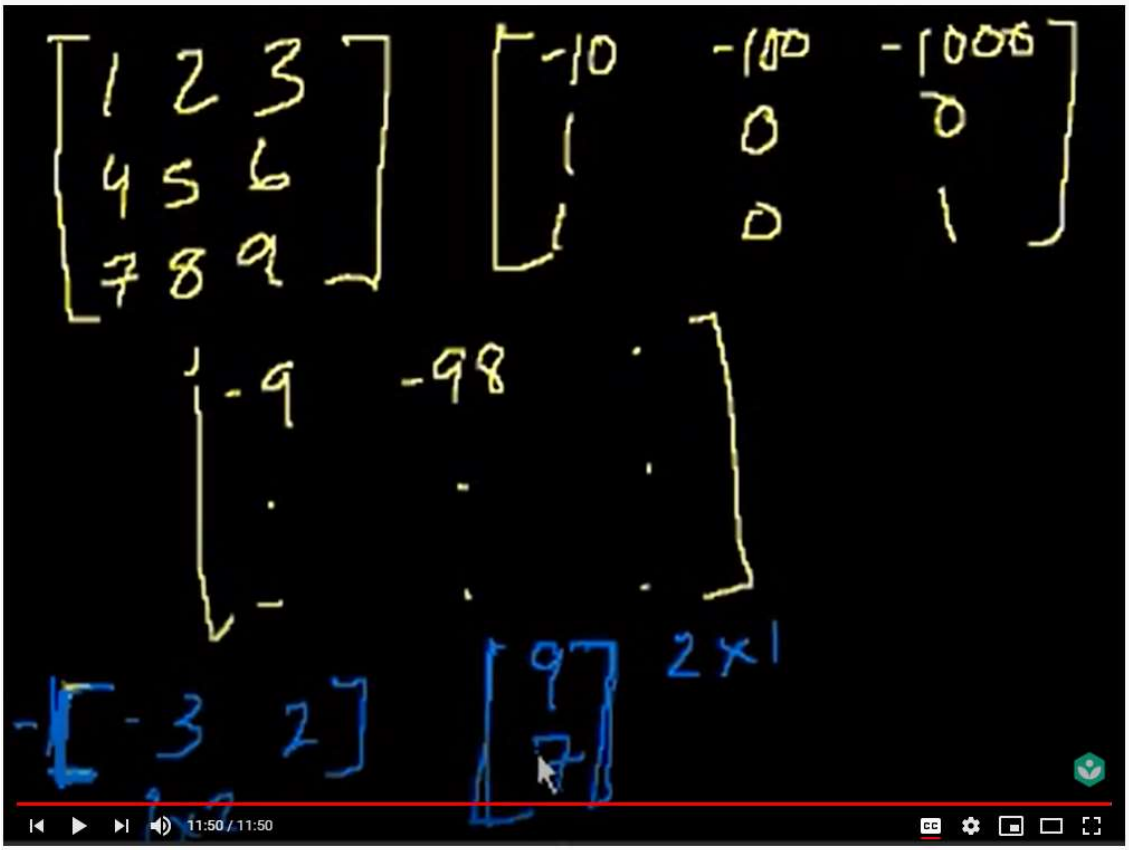
And th

$$\begin{bmatrix} \vec{a}_1^T \\ \vec{a}_2^T \end{bmatrix} \vec{x} = \begin{bmatrix} \vec{a}_1 \cdot \vec{x} \\ \vec{a}_2 \cdot \vec{x} \end{bmatrix}$$


But you can interpret it as a
weighted combination, or a


all just really human-defined
conventions, and then later


minus 1, and 0 plus 5 is 5.


11:50 / 11:50

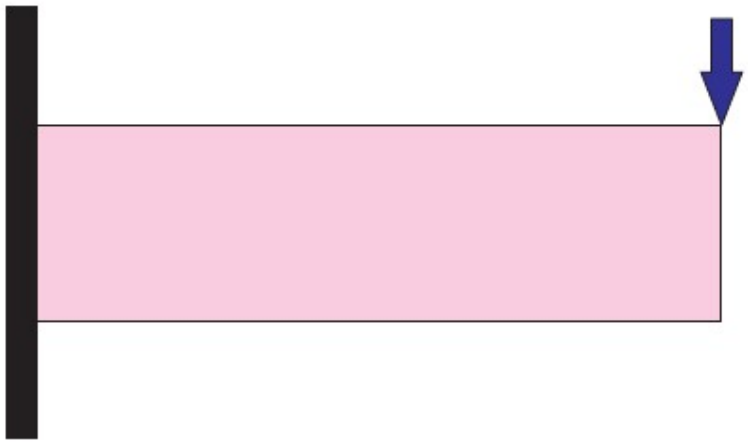
Attempt Coding:

Steps to Finite Element Analysis

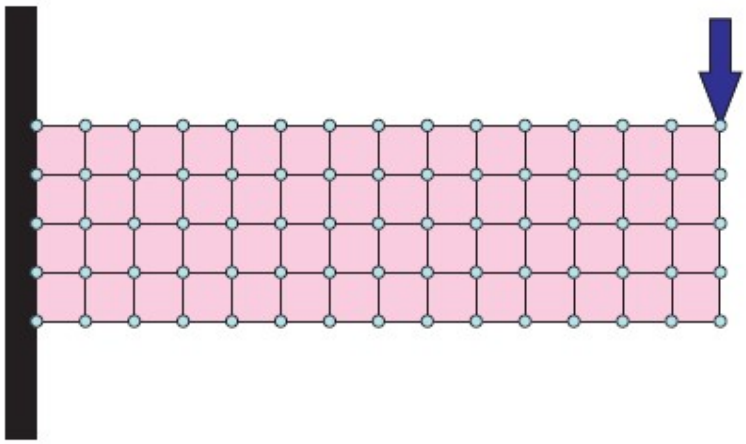
In general, the following steps are required to be performed in any finite element analysis.

1. Discretize the Continuum
2. Generate Element Matrices: Stiffness matrix; load vector
3. Assemble Element Matrices: Form global Stiffness matrix
4. Apply Boundary Conditions
5. Solve Equations and Calculate Deformations
6. Calculate Strains
7. Calculate Stresses

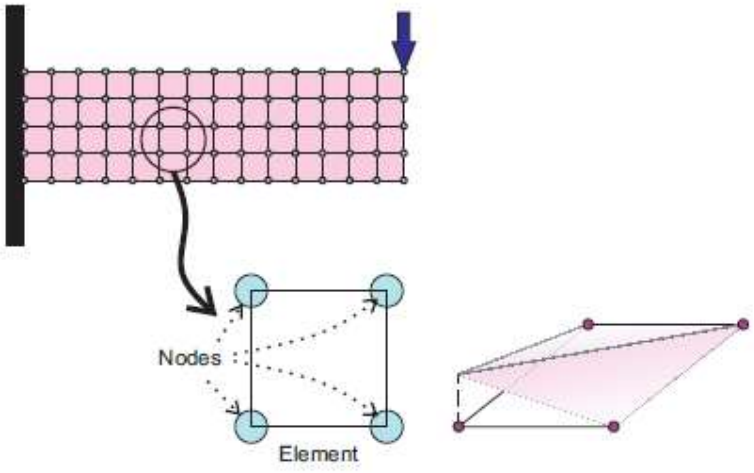
Discretization of the continuum involves dividing the given problem domain into number of subdomains called "elements". Each element is bounded and defined by imaginary points called "nodes". For example, consider a beam as shown in figure below:



To analyse the beam by finite element method, we must subdivide the domain (discretize) it into number of finite elements. The type of element is a matter of choice that the analyst must select based on his engineering experience and judgement. If we were to discretize the domain by quadrilateral elements, it could look like:

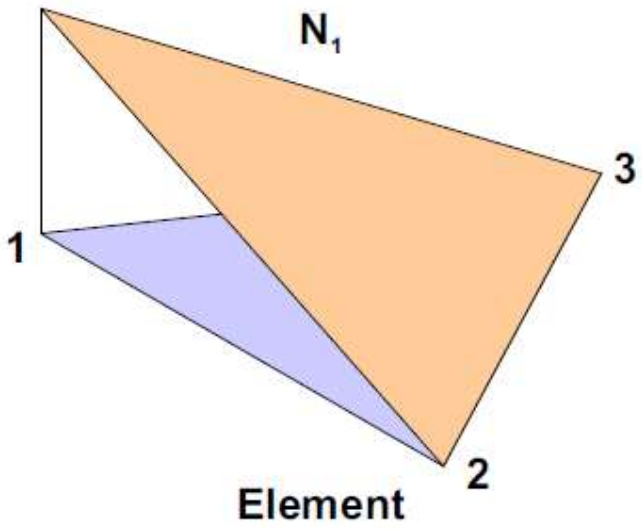


An element is defined by nodes. In the domain, each element can be seen separately as:

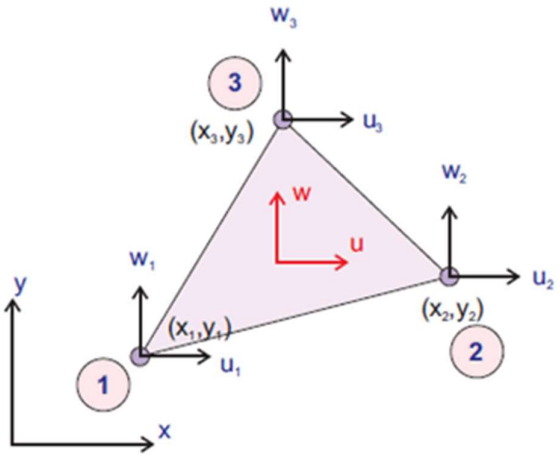


Discretization or "meshing" as more commonly known, is a huge science in itself and is considered to be out of scope for this article.

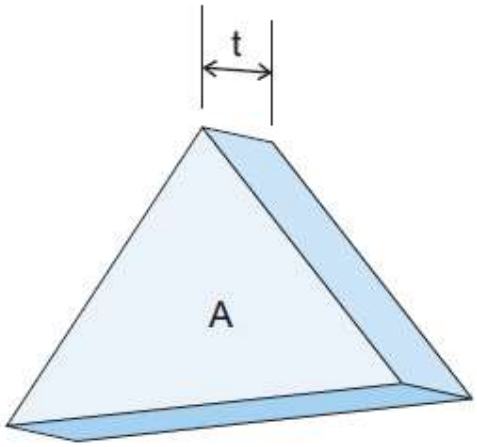
Each element assumes a displacement field over the domain. Based on this assumption, the stiffness matrices of the element get generated. A 3-noded triangular element having linear variation of deformations over the domain has been assumed here. The element has linear shape functions having horizontal and vertical translation at each node. The shape function (at node 1) and degrees of freedom at each node may be depicted as:



The element is very basic and simplistic, but suffices to provide for the basic understanding. The element has 2 degrees of freedom per node and hence produces a 6x6 stiffness matrix. Since the deformations are linear, the strains and consequently the stresses are constant. Hence, this element is commonly known as the Constant Strain Triangle (CST). The stiffness matrix may be derived using the minimization of potential energy or Rayleigh-Ritz Method or any other variational method. The readers are requested to refer to more fundamental references on the subject. For completeness, the key equations for deriving the stiffness matrix of the element are presented here.



- 3 nodes per element
- 2 dofs per node
  - translation in x direction
  - translation in y direction
- Total 6 degrees of freedoms (DOFs) per element
  - 6x6 stiffness matrix



Each node has two degrees of freedom - u and w, or commonly termed as u and v. Shape functions approximate the variation of a field over the domain of the element. Thus, at any point on the element, the displacements may be obtained as:

$$u = \begin{Bmatrix} u(x,y) \\ v(x,y) \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix}$$

$$u_{2x1} = N_{2x6}\ d_{6x1}$$

$$u(x,y) = N_1 u_1 + N_2 u_2 + N_3 u_3$$

$$v(x,y) = N_1 v_1 + N_2 v_2 + N_3 v_3$$

$$N = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix}$$

where, [N] is the Shape Function matrix and the shape functions are given by:

$$N_1 = \frac{a_1 + b_1 x + c_1 y}{2A}$$

$$N_2 = \frac{a_2 + b_2 x + c_2 y}{2A}$$

$$N_3 = \frac{a_3 + b_3 x + c_3 y}{2A}$$

where, A is the area of triangular element, which may be determined as half of the determinant of the Jacobian matrix or as:

$$A = \textbf{area of triangle} = \frac{1}{2}\det\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix}$$

Usual procedure of generating the stiffness matrix involves the calculation of the Jacobian Matrix **[J]**, Strain-Displacement linking matrix "**[B]**" and finally calculating the stiffness matrix **[Ke]** as:

$$[K_e] = \int_{Vol} B^T D B dv$$

where, as per the usual notations, **[B]** is the strain-displacement linking matrix and **[D]** is the constitutive matrix.

If x_i and y_i are the coordinates of the node i of a given element, x_{ij} is defined as . The approximation of strains and subsequent computation of strain-displacement matrix **[B]** may be carried out as:

$$\varepsilon = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{Bmatrix} = B\, d$$

$$B = \begin{bmatrix} \frac{\partial N_1(x,y)}{\partial x} & 0 & \frac{\partial N_2(x,y)}{\partial x} & 0 & \frac{\partial N_3(x,y)}{\partial x} & 0 \\ 0 & \frac{\partial N_1(x,y)}{\partial y} & 0 & \frac{\partial N_2(x,y)}{\partial y} & 0 & \frac{\partial N_3(x,y)}{\partial y} \\ \frac{\partial N_1(x,y)}{\partial y} & \frac{\partial N_1(x,y)}{\partial x} & \frac{\partial N_2(x,y)}{\partial y} & \frac{\partial N_2(x,y)}{\partial x} & \frac{\partial N_3(x,y)}{\partial y} & \frac{\partial N_3(x,y)}{\partial x} \end{bmatrix}$$

$$[B] = \frac{1}{|J|} \cdot \begin{bmatrix} y_{23} & 0 & y_{31} & 0 & y_{12} & 0 \\ 0 & x_{32} & 0 & x_{13} & 0 & x_{21} \\ x_{32} & y_{23} & x_{13} & y_{31} & x_{21} & y_{12} \end{bmatrix}$$

The constitutive matrix for a 2D membrane problem depends on the assumption of plane stress or plane strain condition. For plane stress condition, the [D] matrix looks like:

$$[D] = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$$

For plane strain condition,

$$[D] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1+\nu & 0 \\ 0 & 0 & \frac{1}{2}-\nu \end{bmatrix}$$

where, E is the modulus of elasticity of the material (isotropic) and v is the Poison's ratio. The Jacobian matrix [J] is obtained as:

$$[J] = \begin{bmatrix} x_{13} & y_{13} \\ x_{23} & y_{23} \end{bmatrix}$$

The area of the triangle is obtained as $A = \frac{1}{2}|J|$

B matrix is in terms of 2D coordinates (x and y). Thus, the product $B^T DB$ gives the area part of the volume integral. The final equation for stiffness matrix becomes:

$$[K_e] = a \cdot t \cdot (B^T DB)$$

where, t is the thickness of the element.

The procedure described above for generation of element stiffness matrix can be shown in code as:

```
''' <summary>
''' Returns the determinant of the Jacobian
''' </summary>
''' <returns></returns>
Private Function getDetJ() As Double
    Return (x13 * y23 - x23 * y13)
End Function

''' <summary>
''' Returns the [B] matrix
''' </summary>
''' <returns></returns>
Public Function getB() As Double(,)
```

```
    Dim B(2, 5) As Double 'initialize a 3x6 array
    Dim detJ As Double = getDetJ()

    B(0, 0) = y23 / detJ
    B(1, 0) = 0.0
    B(2, 0) = x32 / detJ

    B(0, 1) = 0.0
    B(1, 1) = x32 / detJ
    B(2, 1) = y23 / detJ

    B(0, 2) = y31 / detJ
    B(1, 2) = 0.0
    B(2, 2) = x13 / detJ

    B(0, 3) = 0.0
    B(1, 3) = x13 / detJ
    B(2, 3) = y31 / detJ

    B(0, 4) = y12 / detJ
    B(1, 4) = 0.0
    B(2, 4) = x21 / detJ

    B(0, 5) = 0.0
    B(1, 5) = x21 / detJ
    B(2, 5) = y12 / detJ

    Return B
End Function

Private Function getAe() As Double
    Return getDetJ() * 0.5
End Function

Public Function getD() As Double(,)
    Dim D(2, 2) As Double
    Dim c As Double
    If ProblemType = ProblemTypes.PlaneStress Then
        c = ElasticityModulus / (1.0 - (PoissonRatio * PoissonRatio))

        D(0, 0) = 1.0 * c
        D(1, 0) = PoissonRatio * c
        D(2, 0) = 0.0

        D(0, 1) = PoissonRatio * c
        D(1, 1) = 1.0 * c
        D(2, 1) = 0.0

        D(0, 2) = 0.0
        D(1, 2) = 0.0
        D(2, 2) = (1 - PoissonRatio) * 0.5 * c

    Else
        'plane strain
        c = ElasticityModulus / ((1 + PoissonRatio) * (1.0 - 2 * PoissonRatio))
        D(0, 0) = (1.0 - PoissonRatio) * c
        D(1, 0) = PoissonRatio * c
        D(2, 0) = 0.0
```

```
        D(0, 1) = PoissonRatio * c
        D(1, 1) = (1 - PoissonRatio) * c
        D(2, 1) = 0.0

        D(0, 2) = 0.0
        D(1, 2) = 0.0
        D(2, 2) = (0.5 - PoissonRatio) * c
    End If
    Return D
End Function

Public Function getKe() As Double(,)

    Dim B(,) As Double = getB()
    Dim BT(,) As Double = getBT(B)
    Dim D(,) As Double = getD()
    Dim teAe As Double = Thickness * getAe()

    'Compute Ke now
    Dim BTD(,) As Double = MultiplyMatrices(BT, D)
    Dim Ke(,) As Double = MultiplyMatrices(BTD, B)

    'multiply ke by teAe
    Dim i, j As Integer
    For i = 0 To 5
        For j = 0 To 5
            ke(i, j) = ke(i, j) * teAe
        Next
    Next
    Return ke
End Function

Private Function getBT(ByRef B(,) As Double) As Double(,)
    'returns the transpose of [B]
    Dim BT(5, 2) As Double
    Dim i, j As Integer
    For i = 0 To 2
        For j = 0 To 5
            BT(j, i) = B(i, j)
        Next
    Next
    Return BT
End Function

''' <summary>
''' Multiplies matrix a by b and returns the product
''' </summary>
''' <param name="a"></param>
''' <param name="b"></param>
''' <returns></returns>
Private Function MultiplyMatrices(ByRef a(,) As Double, ByRef b(,) As Double) As
Double(,)

    Dim aCols As Integer = a.GetLength(1)
    Dim bCols As Integer = b.GetLength(1)
    Dim aRows As Integer = a.GetLength(0)
    Dim ab(aRows - 1, bCols - 1) As Double
    Dim sum As Double
    For i As Integer = 0 To aRows - 1
```

```
        For j As Integer = 0 To bCols - 1
            sum = 0.0
            For k As Integer = 0 To aCols - 1
                sum += a(i, k) * b(k, j)
            Next
            ab(i, j) += sum
        Next
    Next

    Return ab
End Function
```

Assembly of Equations

The element equations give the properties of the element, not the body being analysed. To analyse the body, all the finite elements must be joined together. This process is mathematically known as assembly of equations. Each element stiffness matrix is assembled to form a global a global matrix. The element matrix is a square matrix of size (Element DOFs x Element DOFs). The stiffness matrix for entire body is of the size (Total DOFs x Total DOFs). Usually, the total degrees of freedom of an assembled body may be in the order of a few hundreds of thousands. We usually try to mesh the domain in such a way that the total degrees of freedom are as few (thousands) as possible, but it is normal to see the degrees of freedom in order of a million in a complex structure. The memory required for a decent 10,000 DOF structure would then be:

$$\text{Number of entries in the matrix} = 10,000 \times 10,000 = 10^8.$$

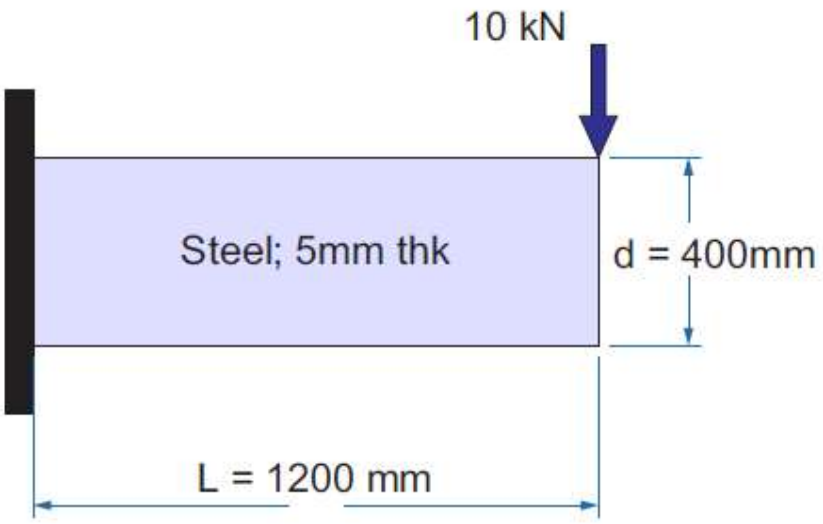
Each entry if saved as a double would require 8 bytes. Thus, the total matrix would require **0.8 GB**. Manipulating a matrix of this size would be time consuming and inhibitive. For decent size structures, we may get a global stiffness matrix that cannot be accommodated in the system memory. However, stiffness matrices are sparse and contain a **lot** of zeros. These zeros can be avoided to be worked on. This allows us to use smart schemes for storing the stiffness matrix. Stiffness matrices are usually stored in the following formats:

1. Banded Matrix
2. Skyline Storage

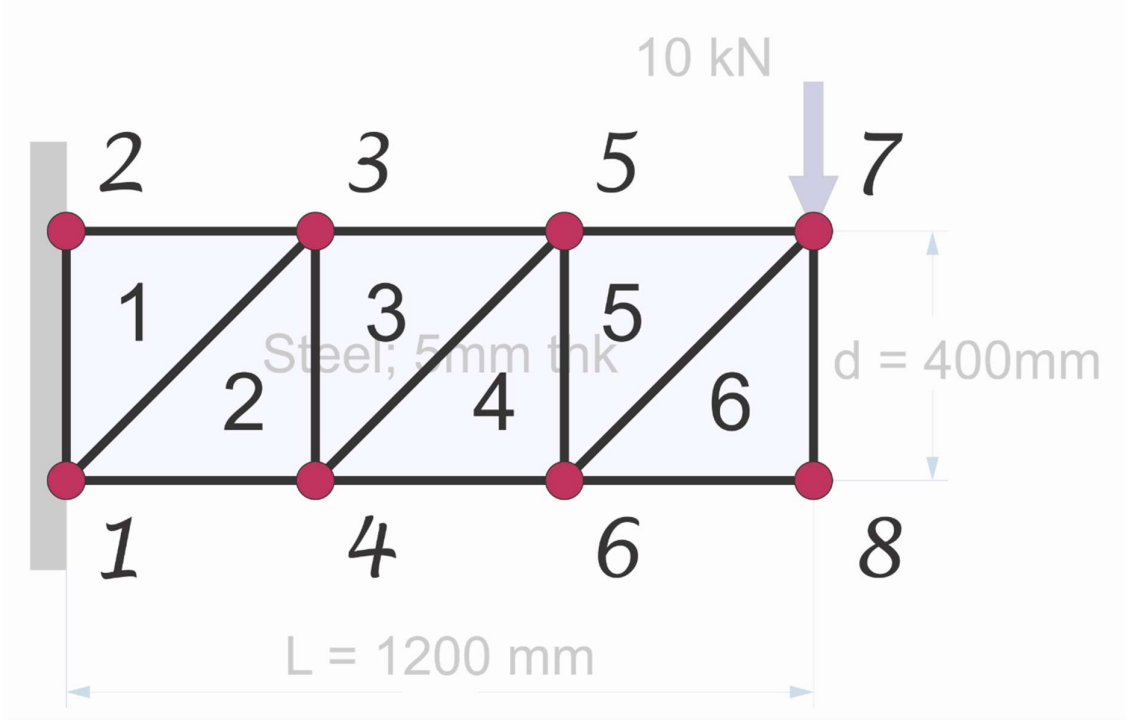
This framework stores the global stiffness matrix in a banded matrix form. The equation solver also must be appropriately modified to handle the type of storage scheme adopted. A gauss elimination solver which works on banded matrices is implemented and given here. The solver implements parallel processing over the factorization block which makes it fast and robust.

The assembly of elements involves placing each elemental matrix in appropriate position in the global matrix based on the degree of freedom. The procedure may be best understood by considering an example.

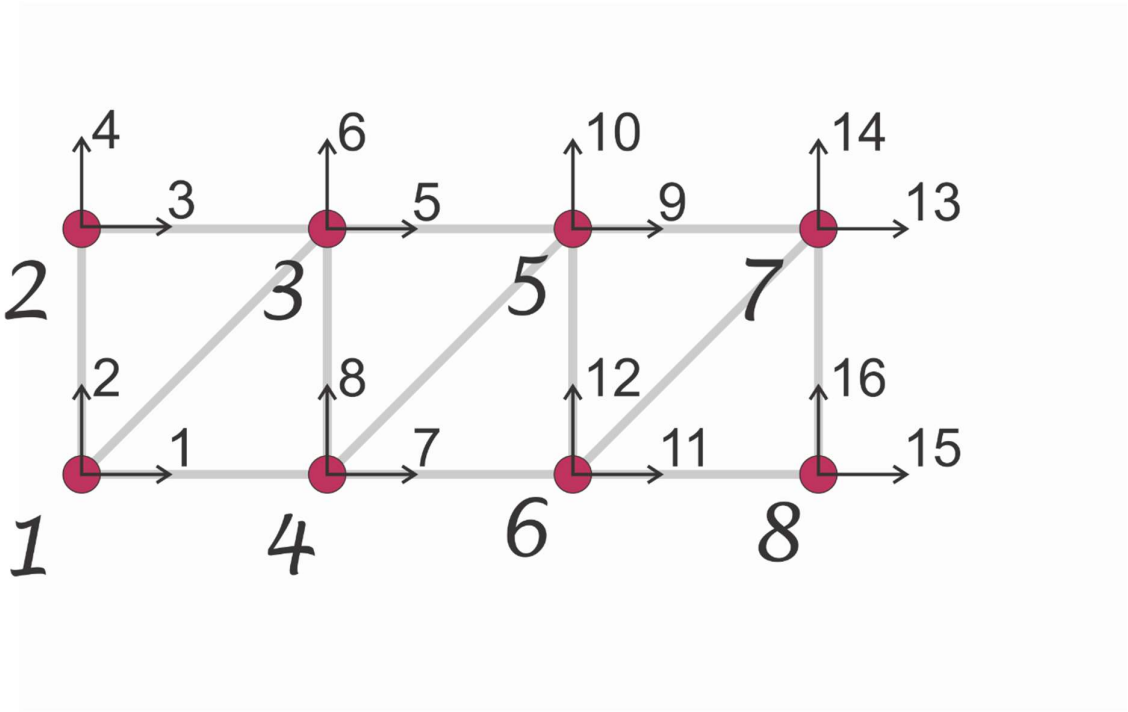
Consider the schematic of a cantilever beam as shown in figure below:



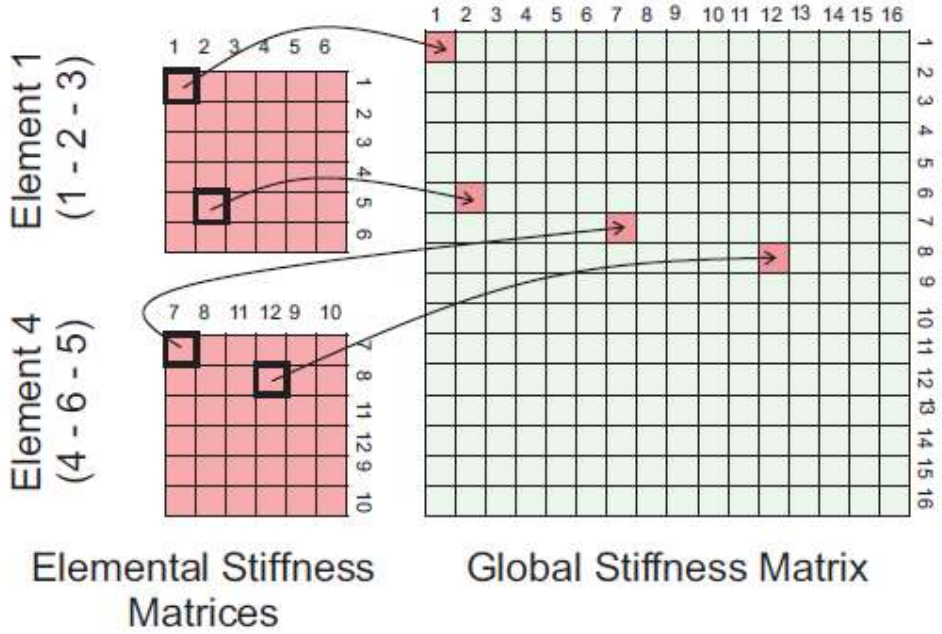
A good mesh would be to discretize this beam into a large number of elements. However, for understanding the procedure, we discretize this beam into a coarse mesh as shown below:



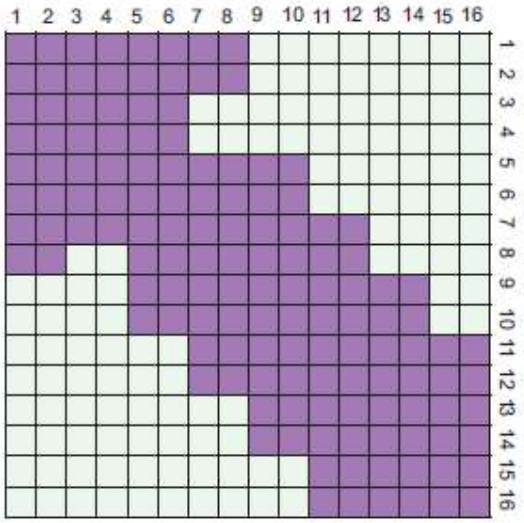
The degrees of freedom at each node are tied by the node number. Thus, the degrees of freedom at each node may be given as:



The element stiffness matrices formed for each element will thus be placed in the global stiffness matrix. An example is shown below:



This produces an assembled global stiffness matrix as:



The algorithm depicted here is implemented as:

```
Private Sub AssembleKg(ByRef Ke(,) As Double, ByRef Kg(,) As Double, ElementNo As
Integer)
    Dim i, j As Integer
    Dim dofs() As Integer = {getDOFx(Elements(ElementNo).Node1 - 1),
                             getDOFy(Elements(ElementNo).Node1 - 1),
                             getDOFx(Elements(ElementNo).Node2 - 1),
                             getDOFy(Elements(ElementNo).Node2 - 1),
                             getDOFx(Elements(ElementNo).Node3 - 1),
                             getDOFy(Elements(ElementNo).Node3 - 1)}

    'Place the upper triangle of the elemental stiffness matrix in the global
    'matrix in proper position
    Dim dofi, dofj As Integer
    For i = 0 To 5 'each dof of the ke
        dofi = dofs(i)
        For j = 0 To 5
            dofj = dofs(j) - dofi
            If dofj >= 0 Then
                Kg(dofi, dofj) = Kg(dofi, dofj) + Ke(i, j)
            End If
        Next
    Next

End Sub

Private Function getDOFx(NodeNo As Integer) As Integer
    Dim nDofsPerNode As Integer = 2
    Return (NodeNo) * nDofsPerNode
End Function
Private Function getDOFy(NodeNo As Integer) As Integer
    Dim nDofsPerNode As Integer = 2
    Return NodeNo * nDofsPerNode + 1
End Function

Private Function getHalfBandWidth() As Integer
```

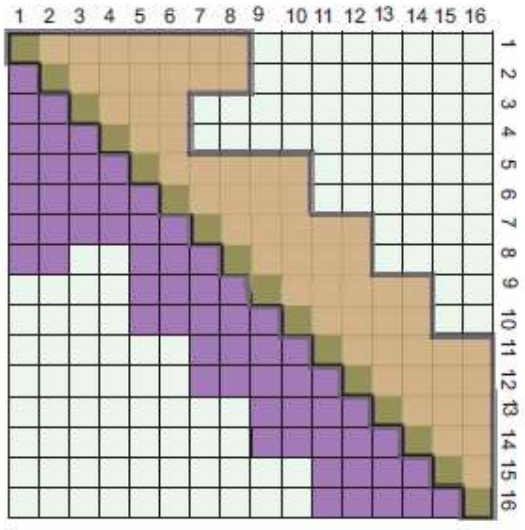
```
    Dim i As Integer
    Dim n(2) As Integer
    Dim MaxDiff As Integer = 0
    Dim diff As Integer
    For i = 0 To NElements - 1
        n(0) = Elements(i).Node1
        n(1) = Elements(i).Node2
        n(2) = Elements(i).Node3
        diff = n.Max - n.Min
        If MaxDiff < diff Then MaxDiff = diff
    Next

    'now we have maxdiff
    'half band width is maxdiff * 2. 2 because there are 2 dofs per node

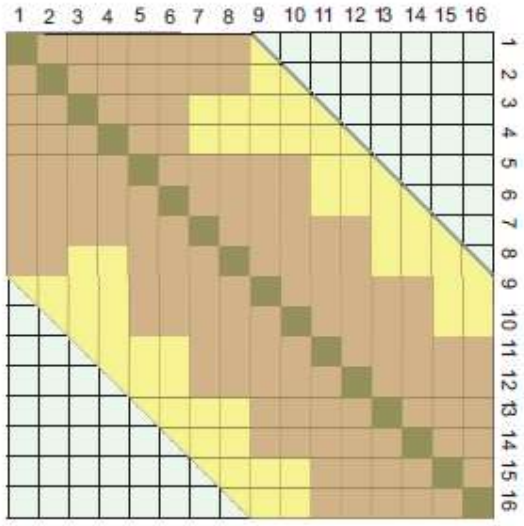
    Dim hbw As Integer = (MaxDiff + 1) * 2
    If hbw > NNodes * 2 Then hbw = NNodes * 2

    Return hbw
End Function
```

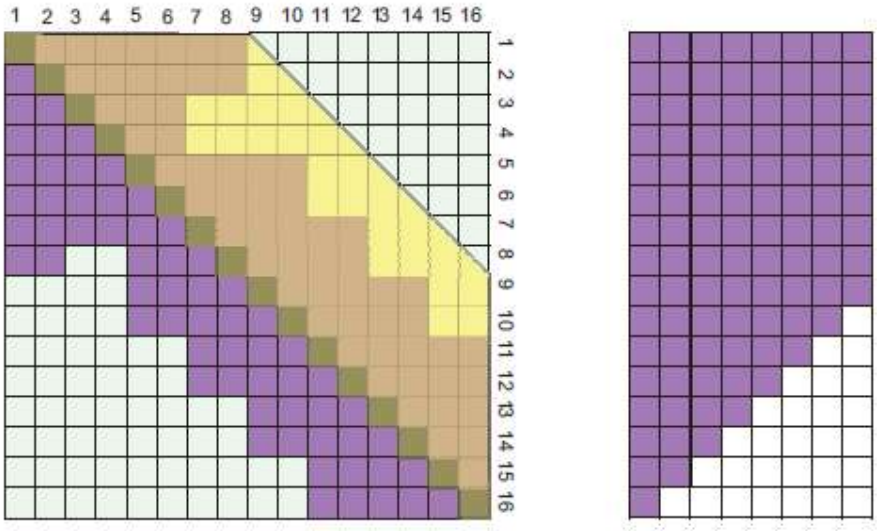
Note that if we swap nodes 1 and 2, the numbering will be more efficient and the resulting band will be more compact. The skyline storage for the matrix shown above would look like:



Skyline storage scheme is obviously more robust, nevertheless, half band storage scheme is adopted here. The dark purple squares show non-zero terms. Thus, avoiding zero terms may allow us to store only the band as shown:



Further, the stiffness matrix for linear problems are symmetric in nature. This allows us to store only a half of the band and still get the correct solution. This calls for a half band storage scheme as adopted in this framework:



The stiffness matrix characterizes the structure / body as if floating in space. The stiffness matrix at this point of time is singular and means that if the structure is given even a slightest disturbance, it will show infinite rigid body deformations. To perform meaningful analysis, we must constrain the body by applying boundary conditions. Boundary conditions could be in the form of loads and supports. While the loads could be of many types, this article considers only nodal forces for example purpose. Further, the supports also could be of many types. Only restraints in x and y direction are considered.

The supports are applied by penalty approach. This involves multiplying the principal diagonal term corresponding with restrained degree of freedom with a large number. Discussion of in-depth mathematics behind the penalty approach is considered to be beyond the scope of this article.

Once the boundary conditions are applied, the next step is solution of equations using gauss elimination and computation of deformations at each degree-of-freedom. This framework performs gauss elimination on banded matrices. After computing the deformations, the elemental deformations can be extracted to compute the strains

from the equation $\{\epsilon\} = [B]\{d\}$ as explained above. The elemental stresses are then calculated from Hooke's law.

Plotting of the elemental strains or stresses require a color map to be defined. A color map defines a variation of color from minimum value to maximum value. For instance, if stress ranges from -20 to 350, the color map will spread the specified colors between -20 and 350. Once the color spreading is done, the index of the color corresponding to the value required can be easily determined by linear interpolation. Here, the color map is defined to vary between Red - Yellow - Green - Cyan - Blue. The code making color map looks like:

```
Private Sub setColors()

        'Dim mainColors() As Color = {Color.Red, Color.Yellow, Color.LightGreen,
        '    Color.Cyan, Color.Blue}
        Dim mainColors() As Color = {Color.Blue, Color.Cyan, Color.LightGreen,
            Color.Yellow, Color.Red}

        Dim nColors As Integer = mainColors.Length
        'this is from top to bottom.

        Dim percentStep As Double = 100 / (nColors - 1)
        Dim Indices(nColors - 1) As Integer 'we will have these many main indices

        Dim i, j As Integer

        For i = 0 To nColors - 1
            Indices(i) = CInt(i * percentStep / 100 * nSegs)
        Next
        Indices(0) = 0
        Indices(Indices.Length - 1) = nSegs
        ReDim Colors(nSegs)
        Dim c() As Color
        For i = 0 To Indices.Length - 2
            c = getSteppedColors(mainColors(i + 1), mainColors(i), Indices(i + 1) -
Indices(i))
            For j = 0 To c.Length - 1
                Colors(Indices(i) + j) = c(j)
            Next
        Next

    End Sub
```

Once the colors are set in the array, the required color may be retrieved simply by:

```
Public Function getColor(Value As Double) As Color

        'value exists somewhere between min and max..
        'we find color between min color and max color corresponding to value and
return
        Dim rangeVal As Double = Max - Min
        Dim dv As Double = Value - Min
        Dim ratio = dv / rangeVal
        Dim colIndex As Integer = CInt(ratio * nSegs)
        Return Colors(colIndex)
End Function
```

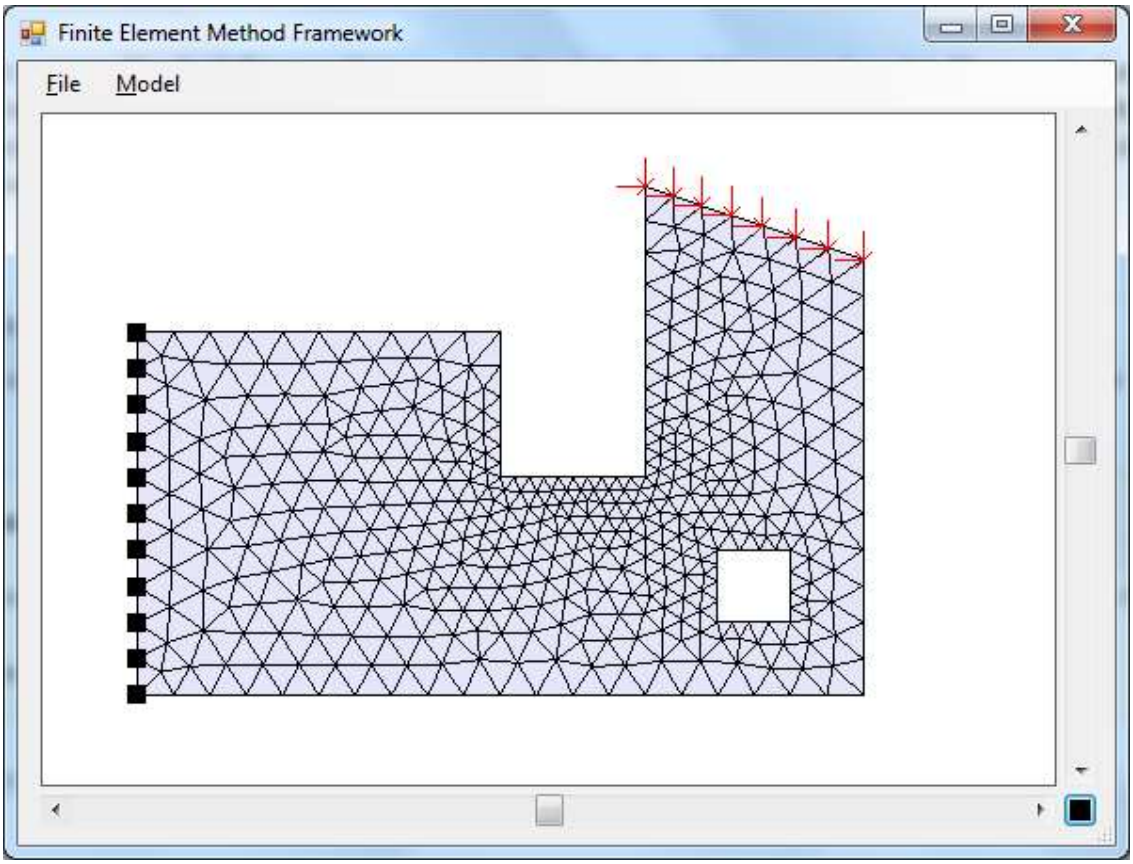
Using the code

The finite element framework is implemented in VB.Net and named btFEM. The main classes implemented are:

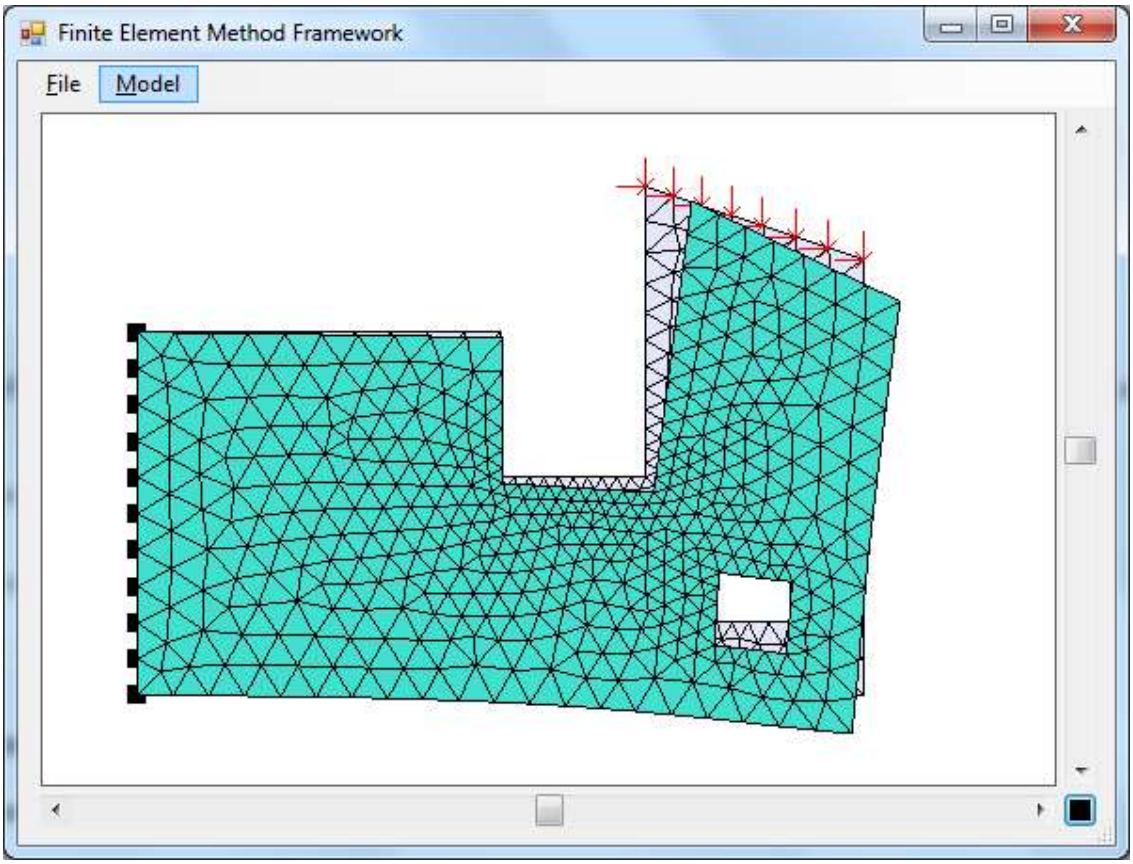
The finite element framework is implemented in VB.Net and named btFEM. The main classes implemented are:

1. CST - Implementation of the Constant Strain Triangle. The class returns **[B]**, **[D]** and **[Ke]**.
2. btGauss - Parallel and serial implementations of the banded gauss solver for solution of equations $[K]d = r$
3. Element - Implementation of element. This is different than CST. Element class contains upper level information such as bounding node numbers for each element, etc.
4. Node - Contains definition for the coordinate of each node
5. PointLoad - Contains definition for the loads in x and y direction for specified node
6. ColorMap - Implements color bar to show stress and strain variation over the body

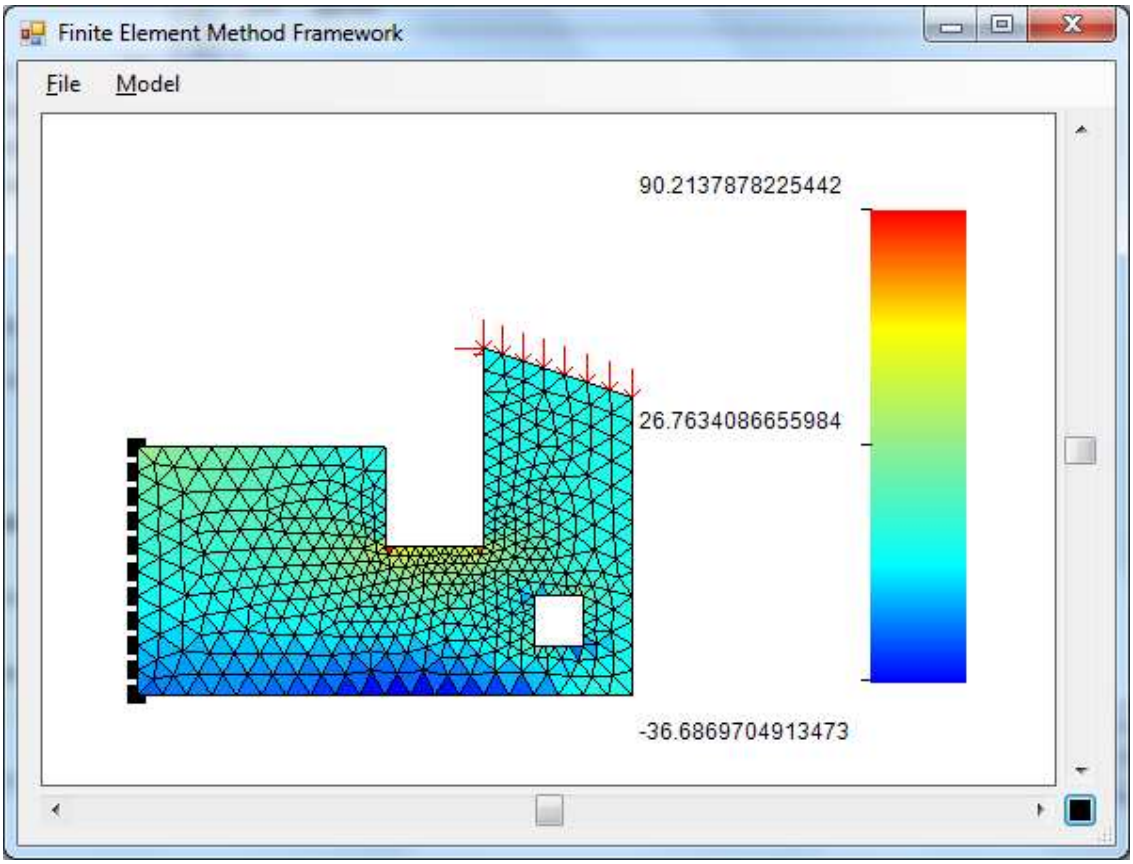
The start-up file: frmbtFem implements a graphical display of the opened model. Please note that a mesh of the model has to be supplied to this program. Left click zooms in and right click zooms out on the model view. Scroll bars pan the model and a button on right bottom of the screen resets zoom to 1 and pan values to 0. An opened finite element model looks like:



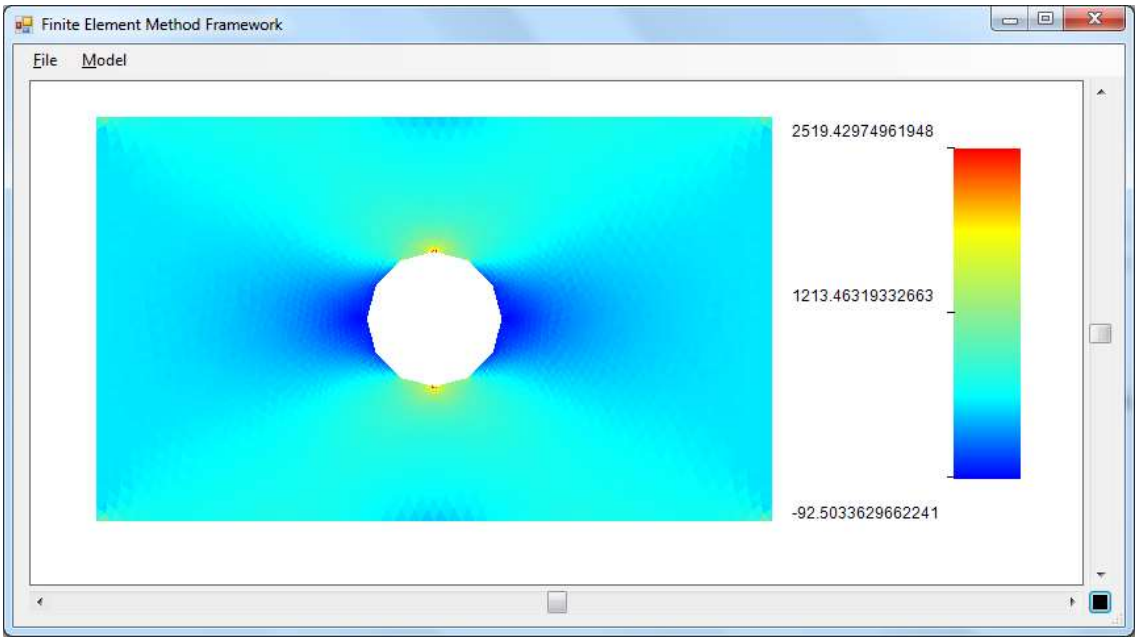
To analyse the structure, click on menu "Model" -> "Analyse". Upon analysis, the program shows the deformed shape as shown below:



Various results may be viewed such as stresses in x direction, stress in y direction and shear stress, or strains in x direction, strains in y direction and shear strains. For example, the plot for stress in x direction is as shown below:



The deformations, stresses and strains may be viewed in a textbox by clicking menu Model -> Results. The framework has been validated by comparing deformations and stresses with standard problems such as cantilever beams with point loads at free end, simply supported beams. The input files for the same have been included with the project files. A typical validation run for a plate with hole is shown in figure below, which shows variation of stress σx:

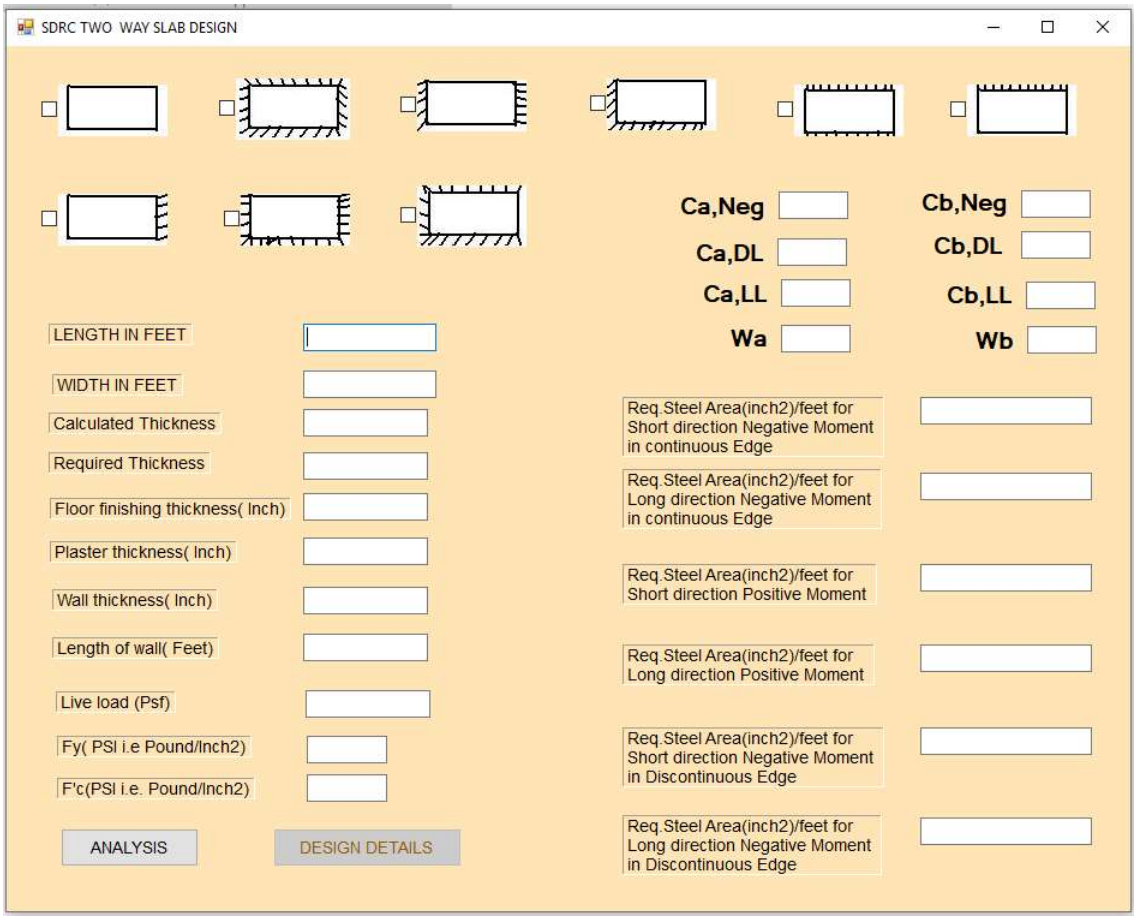


Points of Interest

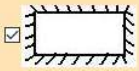
Finite element analysis is a challenging and computationally intense simulation. The core challenge lies in efficient computation of element matrices and solution of equations. Gauss elimination is a versatile technique for solution of equations and can be easily and efficiently modified to work on banded matrices.

Programming the finite element method has been described here. Availability of the complete working program is expected to enhance the understanding of the readers. Robust banded assembler and solver have been included as stand alone classes, which can be easily integrated in any code.

Our Some Product:



SDRC TWO WAY SLAB DESIGN



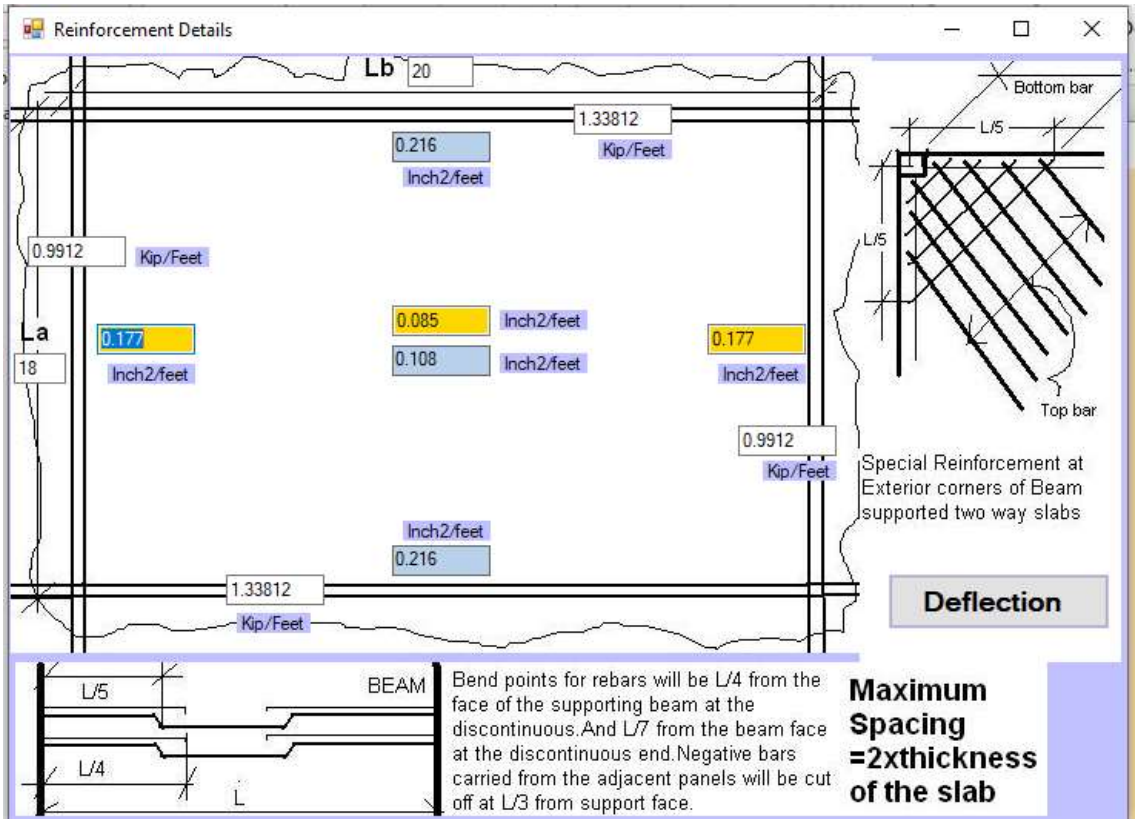
Ca,Neg	0.055	Cb,Neg	0.037
Ca,DL	0.022	Cb,DL	0.014
Ca,LL	0.034	Cb,LL	0.022
Wa	0.6	Wb	0.4

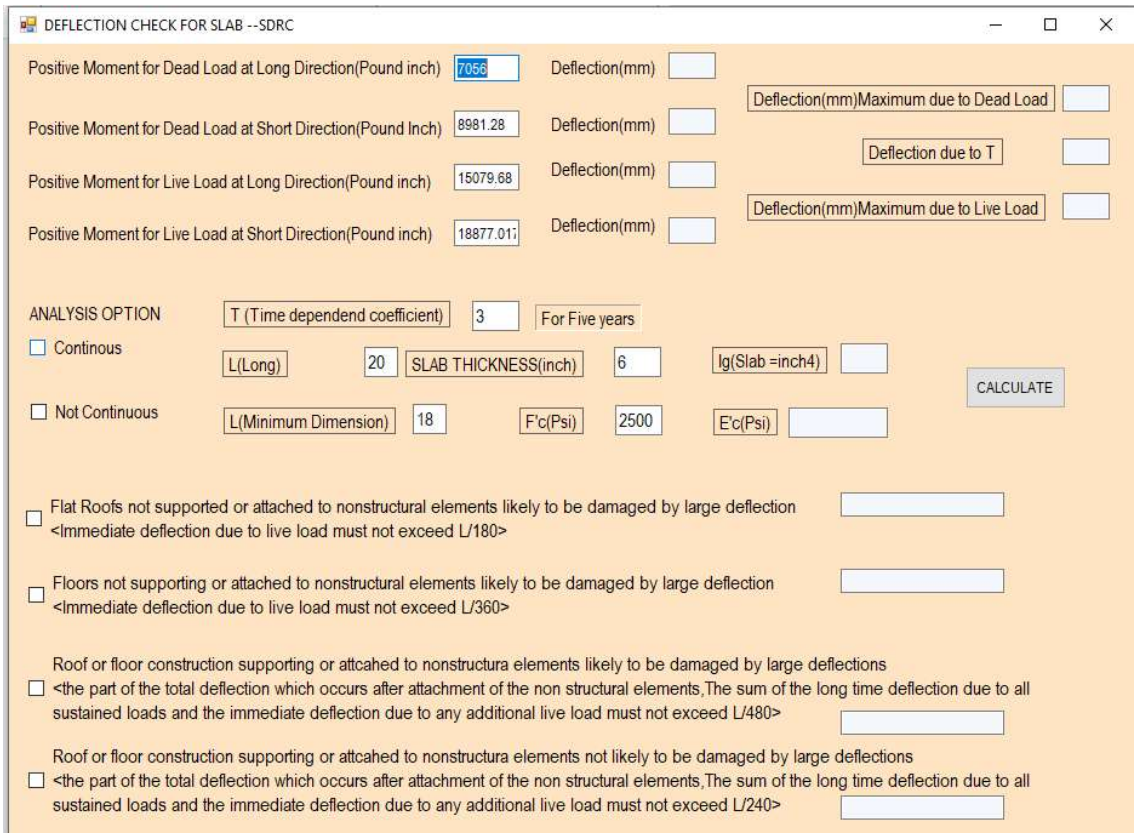
LENGTH IN FEET	20
WIDTH IN FEET	18
Calculated Thickness	5.066666666666
Required Thickness	6
Floor finishing thickness(Inch)	
Plaster thickness(Inch)	
Wall thickness(Inch)	
Length of wall(Feet)	
Live load (Psf)	84
Fy(PSI i.e Pound/Inch2)	60000
F'c(PSI i.e. Pound/Inch2)	2500

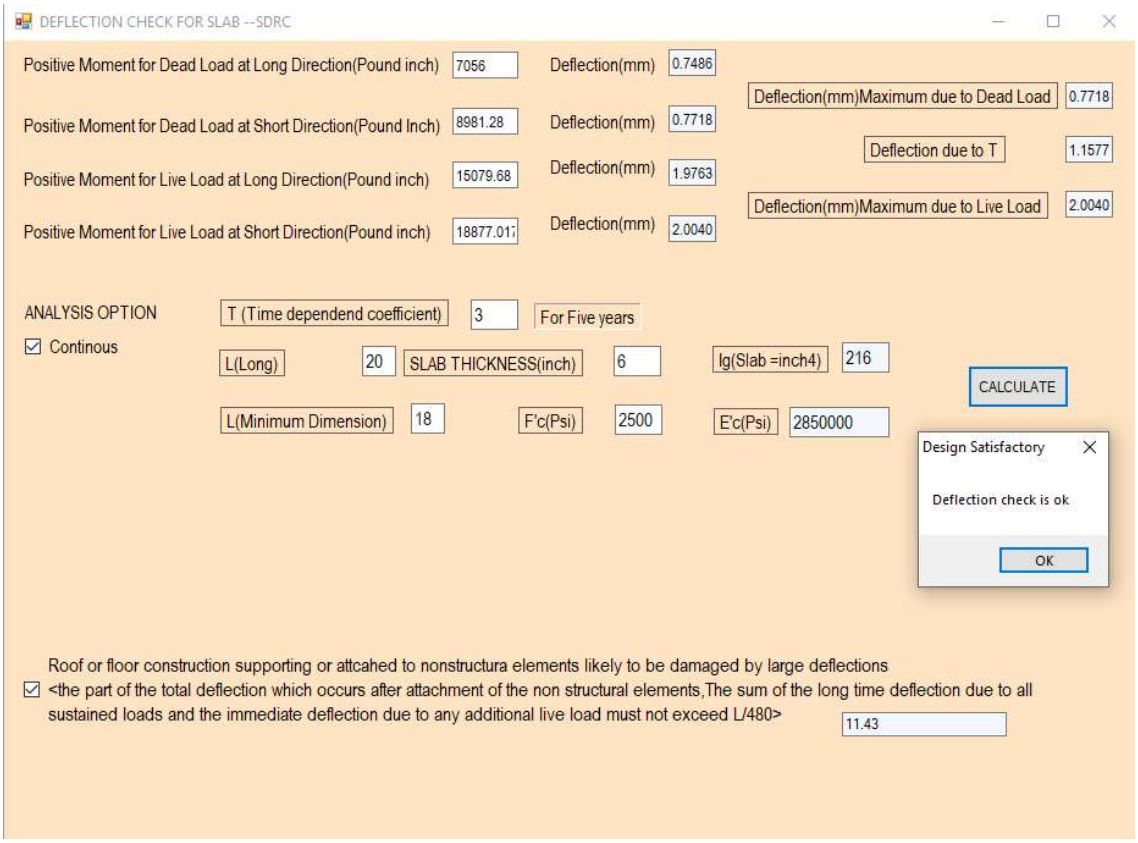
ANALYSIS

DESIGN DETAILS

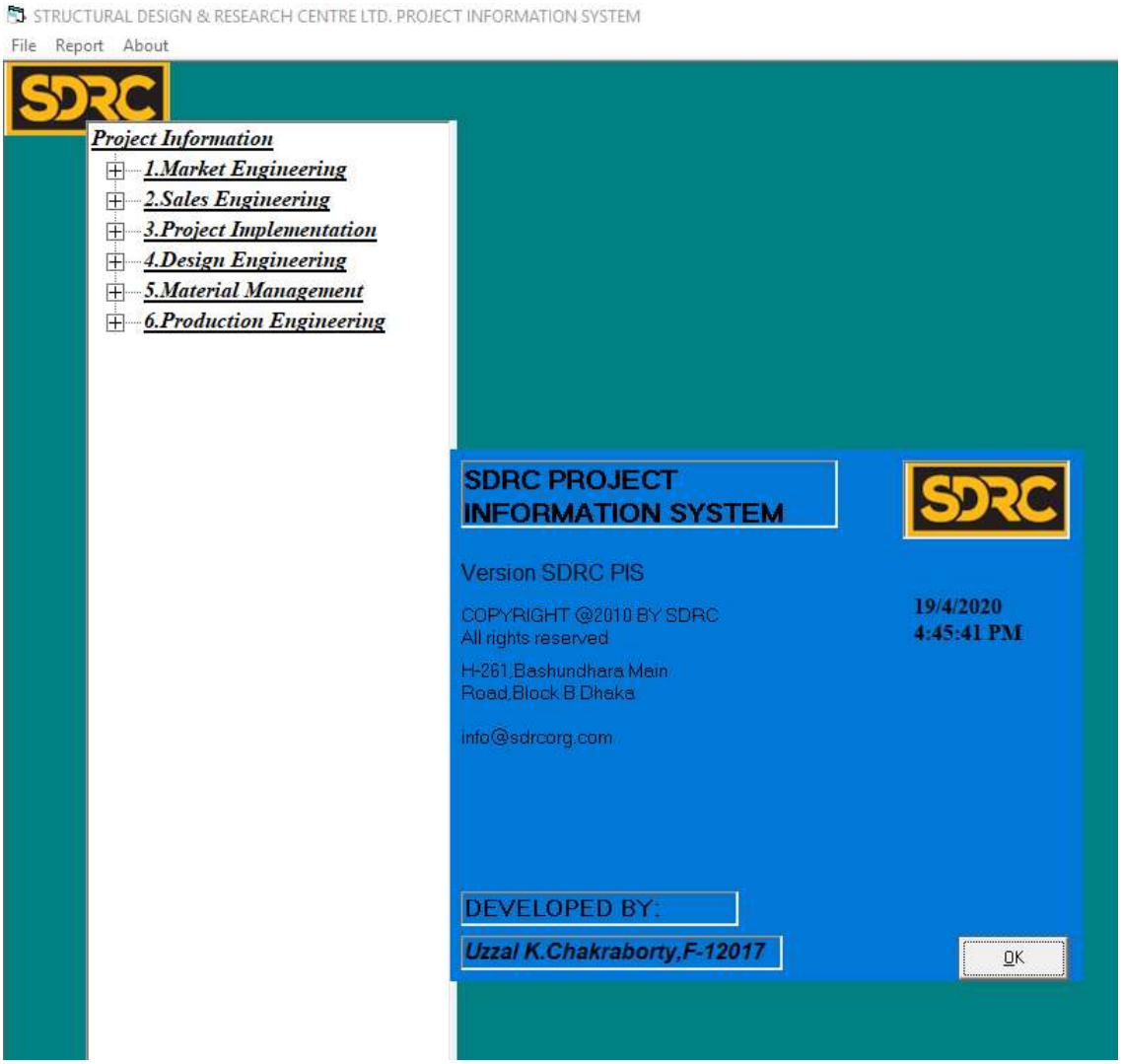
Req.Steel Area(inch2)/feet for Short direction Negative Moment in continuous Edge	0.216
Req.Steel Area(inch2)/feet for Long direction Negative Moment in continuous Edge	0.177
Req.Steel Area(inch2)/feet for Short direction Positive Moment	0.108
Req.Steel Area(inch2)/feet for Long direction Positive Moment	0.085
Req.Steel Area(inch2)/feet for Short direction Negative Moment in Discontinuous Edge	0
Req.Steel Area(inch2)/feet for Long direction Negative Moment in Discontinuous Edge	0

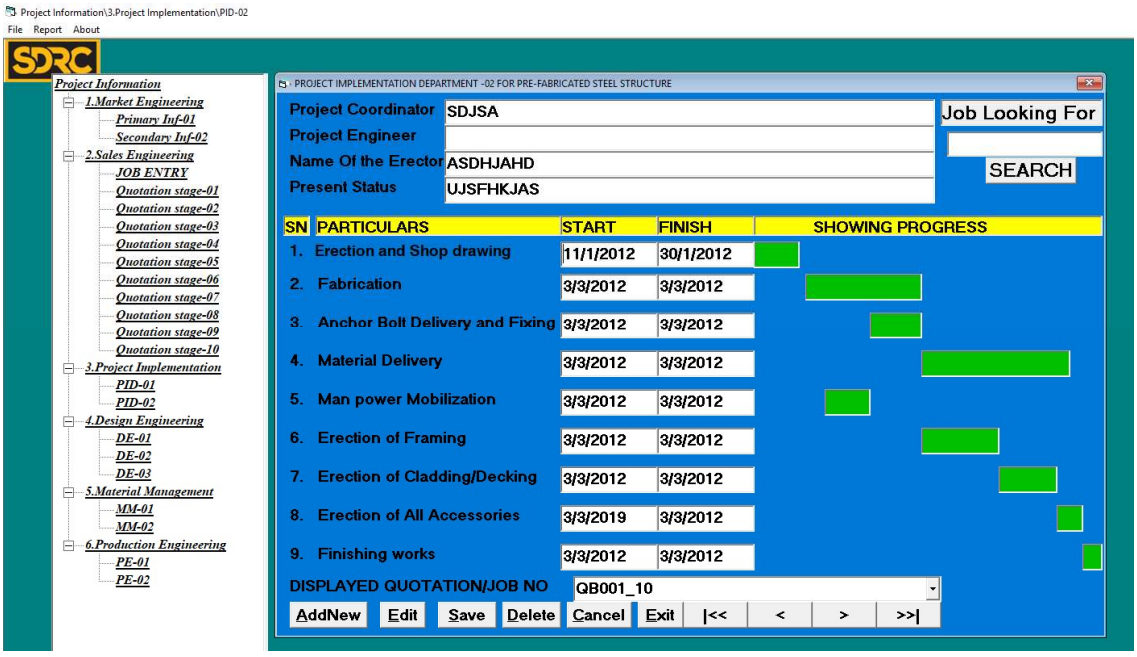

Reinforcement Details
Lb
20
1.33812
Kip/Feet
0.216
Inch2/feet
0.9912
Kip/Feet
La
18
0.177
Inch2/feet
0.085
Inch2/feet
0.108
Inch2/feet
0.177
Inch2/feet
0.9912
Kip/Feet
Inch2/feet
0.216
1.33812
Kip/Feet
Bottom bar
L/5
L/5
Top bar
Special Reinforcement at Exterior corners of Beam supported two way slabs
Deflection
L/5
BEAM
L/4
L
Bend points for rebars will be L/4 from the face of the supporting beam at the discontinuous.And L/7 from the beam face at the discontinuous end.Negative bars carried from the adjacent panels will be cut off at L/3 from support face.
Maximum Spacing =2xthickness of the slab


DEFLECTION CHECK FOR SLAB --SDRC
Positive Moment for Dead Load at Long Direction(Pound inch)
7056
Deflection(mm)
Positive Moment for Dead Load at Short Direction(Pound Inch)
8981.28
Deflection(mm)
Positive Moment for Live Load at Long Direction(Pound inch)
15079.68
Deflection(mm)
Positive Moment for Live Load at Short Direction(Pound inch)
18877.017
Deflection(mm)
Deflection(mm)Maximum due to Dead Load
Deflection due to T
Deflection(mm)Maximum due to Live Load
ANALYSIS OPTION
Continous
Not Continuous
T (Time dependend coefficient)
3
For Five years
L(Long)
20
SLAB THICKNESS(inch)
6
Ig(Slab =inch4)
L(Minimum Dimension)
18
F'c(Psi)
2500
E'c(Psi)
CALCULATE
Flat Roofs not supported or attached to nonstructural elements likely to be damaged by large deflection
<Immediate deflection due to live load must not exceed L/180>
Floors not supporting or attached to nonstructural elements likely to be damaged by large deflection
<Immediate deflection due to live load must not exceed L/360>
Roof or floor construction supporting or attcahed to nonstructura elements likely to be damaged by large deflections
<the part of the total deflection which occurs after attachment of the non structural elements,The sum of the long time deflection due to all sustained loads and the immediate deflection due to any additional live load must not exceed L/480>
Roof or floor construction supporting or attcahed to nonstructura elements not likely to be damaged by large deflections
<the part of the total deflection which occurs after attachment of the non structural elements,The sum of the long time deflection due to all sustained loads and the immediate deflection due to any additional live load must not exceed L/240>


DEFLECTION CHECK FOR SLAB --SDRC
Positive Moment for Dead Load at Long Direction(Pound inch) 7056
Deflection(mm) 0.7486
Positive Moment for Dead Load at Short Direction(Pound Inch) 8981.28
Deflection(mm) 0.7718
Positive Moment for Live Load at Long Direction(Pound inch) 15079.68
Deflection(mm) 1.9763
Positive Moment for Live Load at Short Direction(Pound inch) 18877.01
Deflection(mm) 2.0040
Deflection(mm)Maximum due to Dead Load 0.7718
Deflection due to T 1.1577
Deflection(mm)Maximum due to Live Load 2.0040
ANALYSIS OPTION
Continous
T (Time dependend coefficient) 3
For Five years
L(Long) 20
SLAB THICKNESS(inch) 6
Ig(Slab =inch4) 216
L(Minimum Dimension) 18
F'c(Psi) 2500
E'c(Psi) 2850000
CALCULATE
Design Satisfactory
Deflection check is ok
OK
Roof or floor construction supporting or attcahed to nonstructura elements likely to be damaged by large deflections
<the part of the total deflection which occurs after attachment of the non structural elements,The sum of the long time deflection due to all sustained loads and the immediate deflection due to any additional live load must not exceed L/480>
11.43

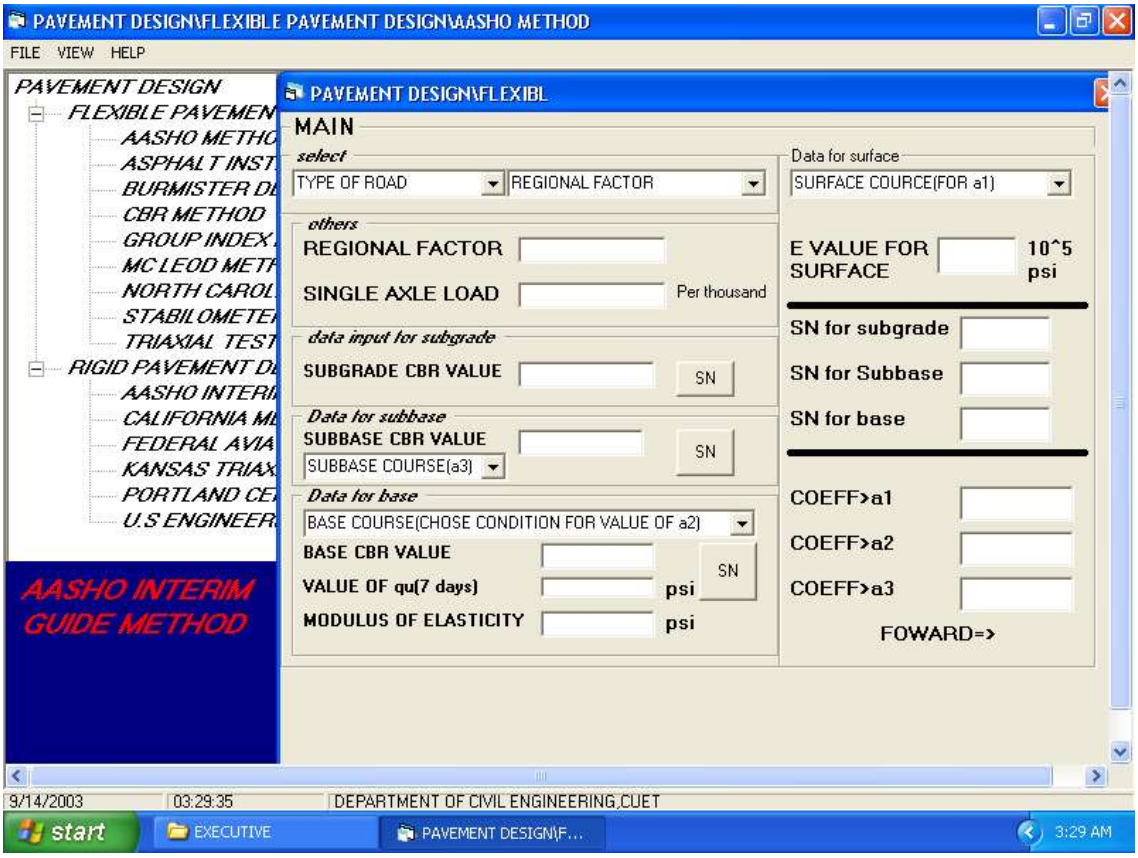

DIRET DESIGN METHOD SDRC PRODUCT
SLAB WITH BEAMS BETWEEN ALL SUPPORTS
INPUT
SHORT CLEARSPAN(in Feet) 20
LONG CLEAR SPAN(in Feet) 25
F'c(PSI) 4000
Fy(PSI) 60000
LIVE LOAD IN PSF 125
PLEASE ASSUME SLAB THICKNESS 7
PLEASE ASSUME BEAM DIMENSION (SMALL) in Inch 14
PLEASE ASSUME BEAM DIMENSION (Large) 20
Ecb(MODULUS OF ELASTICITY FOR BEAM) 1
Ecs(MODULUS OF ELASTICITY FOR SLAB) 1
CALCULATE
OUTPUT
MOMENT CHART
LOCATION
Mu,ft-Kips
b,INCH
d,INCH
Mu*12/b ft-Kips/ft
LDTWO 1/2 COL STRIP
EXT.NEG 10.7 106 1.2
POSITIVE 32.9 106 5.5 3.7
INT.NEG 40.5 106 4.6
MIDDLE STRIP
EXT.NEG 4.7 120 0.5
POSITIVE 51.5 120 5.2
INT.NEG 63.3 120 6.3
SDEX1/2
NEGATIVE 12.7 53 2.9
POSITIVE 6.8 53 6 1.5
MIDDLE
NEGATIVE 80.1 180 5.3
POSITIVE 43.1 180 2.9
INT 1/2 COL STR
NEGATIVE 12.1 53 2.7
POSITIVE 6.5 53 1.5
SHORT DIRECTION
INTERIOR SLAB BEAM STRIP FOR SHORT DIR..
NEGATIVE POSITIVE
BEAM MOMENT 137 73.8
COLUMN STRIP SLAB MOMENT 24.2 13
MIDDLE STRIP SLAB MOMENT 80.1 43.1
EXTERIOR SLAB BEAM STRIP FOR SHORT DIR..
BEAM MOMENT 71.7 38.6
COLUMN STRIP SLAB MOMENT 12.7 6.8
MIDDLE STRIP SLAB MOMENT 41.9 22.6
LONG DIRECTION
EXTERIOR NEGATIVE
POSITIVE
INTERIOR NEGATIVE
BEAM MOMENT 60.7 186.7 229.3
COLUMN STRIP SLAB MOMENT 10.7 32.9 40.5
MIDDLE STRIP SLAB MOMENT 4.7 51.5 63.3

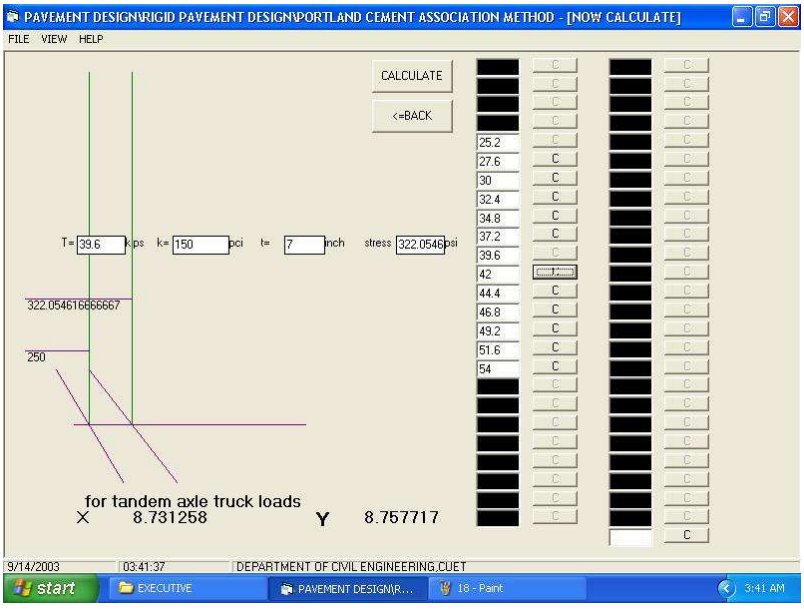

STRUCTURAL DESIGN & RESEARCH CENTRE LTD. PROJECT INFORMATION SYSTEM
File Report About
Project Information
1.Market Engineering
2.Sales Engineering
3.Project Implementation
4.Design Engineering
5.Material Management
6.Production Engineering
SDRC PROJECT INFORMATION SYSTEM
Version SDRC PIS
COPYRIGHT @2010 BY SDRC
All rights reserved
19/4/2020
4:45:41 PM
H-261,Bashundhara Main Road,Block B Dhaka
info@sdrcorg.com
DEVELOPED BY:
Uzzal K.Chakraborty,F-12017
OK

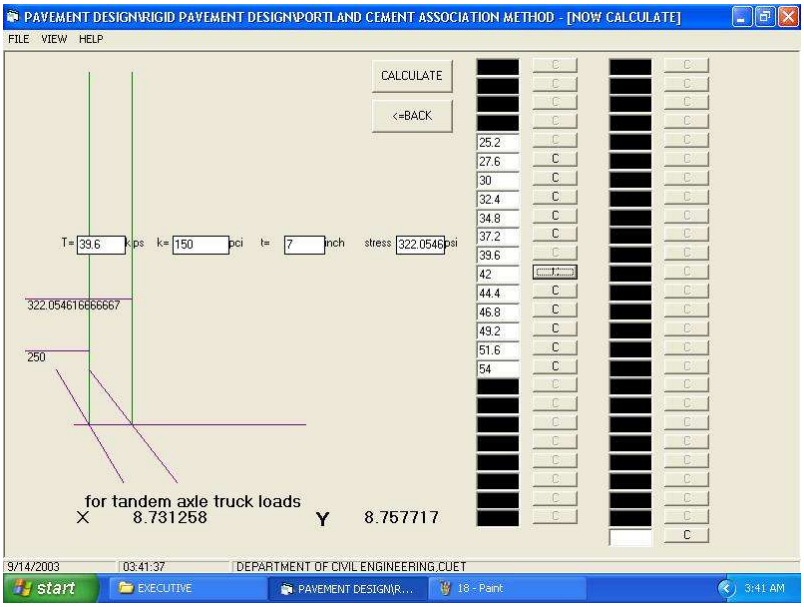

Project Information\3.Project Implementation\PID-02
File Report About
Project Information
1.Market Engineering
Primary Inf-01
Secondary Inf-02
2.Sales Engineering
JOB ENTRY
Quotation stage-01
Quotation stage-02
Quotation stage-03
Quotation stage-04
Quotation stage-05
Quotation stage-06
Quotation stage-07
Quotation stage-08
Quotation stage-09
Quotation stage-10
3.Project Implementation
PID-01
PID-02
4.Design Engineering
DE-01
DE-02
DE-03
5.Material Management
MM-01
MM-02
6.Production Engineering
PE-01
PE-02
PROJECT IMPLEMENTATION DEPARTMENT -02 FOR PRE-FABRICATED STEEL STRUCTURE
Project Coordinator SDJSA
Project Engineer
Name Of the Erector ASDHJAHD
Present Status UJSFHKJAS
Job Looking For
SEARCH
SN PARTICULARS START FINISH SHOWING PROGRESS
1. Erection and Shop drawing 11/1/2012 30/1/2012
2. Fabrication 3/3/2012 3/3/2012
3. Anchor Bolt Delivery and Fixing 3/3/2012 3/3/2012
4. Material Delivery 3/3/2012 3/3/2012
5. Man power Mobilization 3/3/2012 3/3/2012
6. Erection of Framing 3/3/2012 3/3/2012
7. Erection of Cladding/Decking 3/3/2012 3/3/2012
8. Erection of All Accessories 3/3/2019 3/3/2012
9. Finishing works 3/3/2012 3/3/2012
DISPLAYED QUOTATION/JOB NO QB001_10
AddNew Edit Save Delete Cancel Exit |<< < > >>|

HIGHWAY PAVEMENT DESIGN

Developed by
BY
Engr.Uzzal Kanti Chakraborty
Principal Consultant
Structural Design & Research Centre







(Aiming to do all in a click!!)