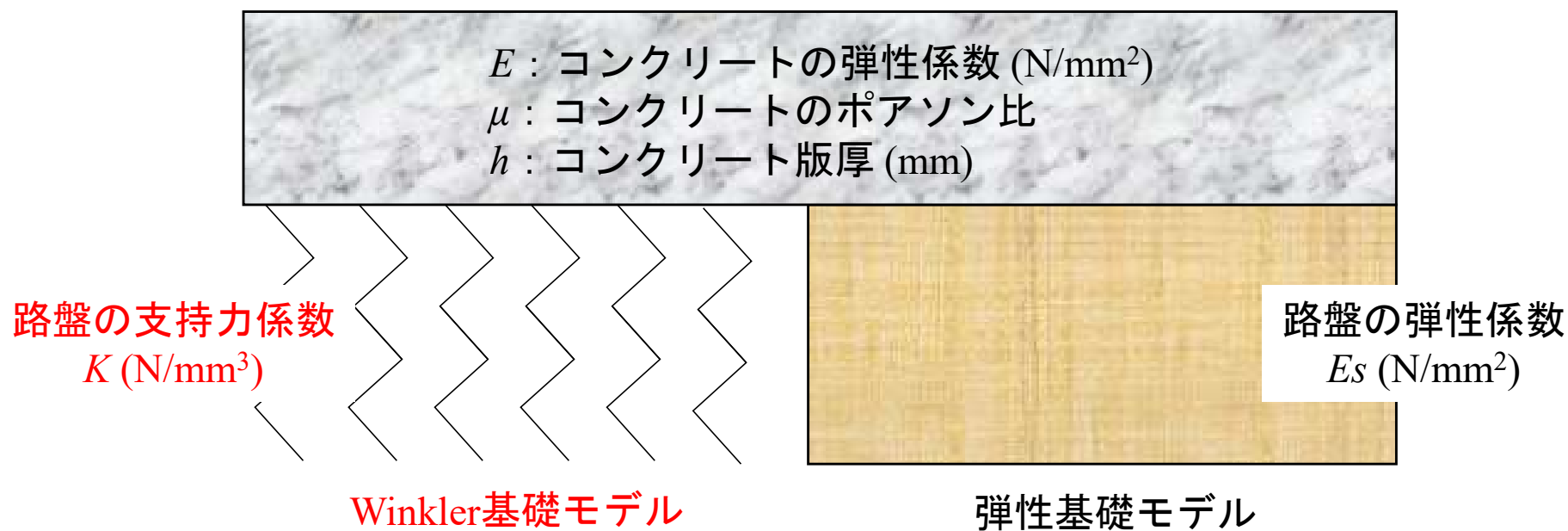
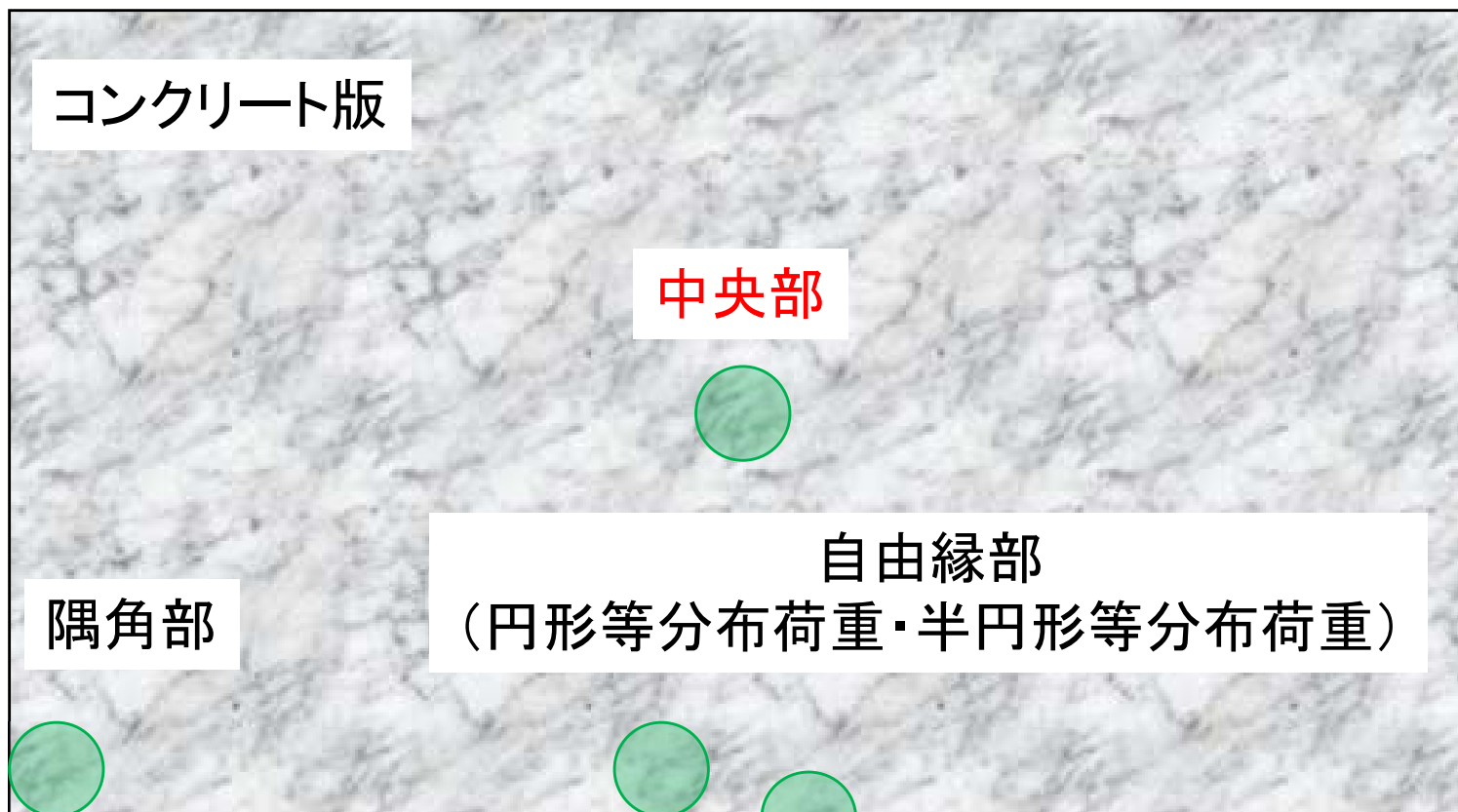


コンクリート舗装の Westergaard中央部載荷プログラム の改良

空港研究部 空港新技術研究官
坪川 将丈



概要・背景

【概要】

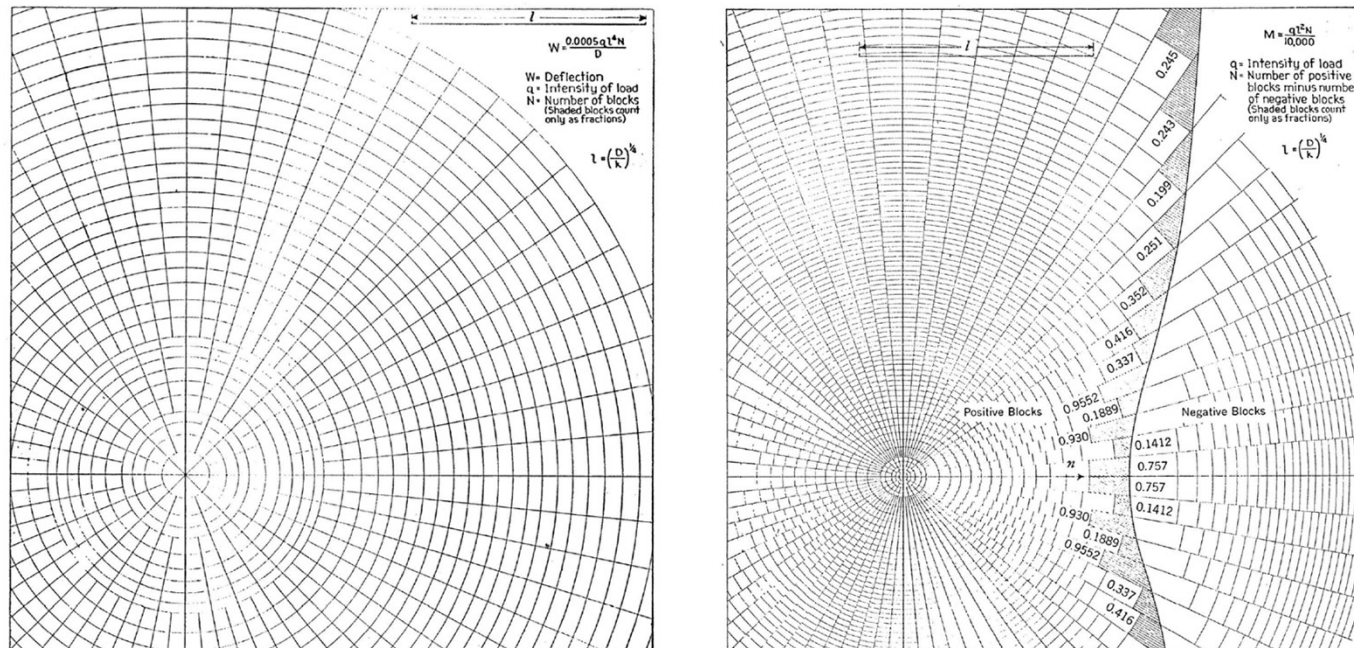
1977年に運輸省港湾技術研究所 滑走路研究室の福手勤(現:東洋大学名誉教授)が作成した, Winkler基礎モデルのWestergaard中央部載荷プログラムをVBAで再構築した.

【背景】

- 現プログラムはFortranで作成されている. コンクリート舗装のたわみ・応力を非常に簡便に解析することが可能.
- 今後, OS等の変更等により動作しなくなる／コンパイラによっては動作しない可能性がゼロではない.
- 楕円形荷重のみが対象.
- 任意の多数の点の応力・たわみを出力する際に若干手間がかかる.
- 円形荷重・矩形荷重も考慮できるよう, またEXCELで動作するよう, 新プログラムをVBAで構築した.

現プログラムの概要

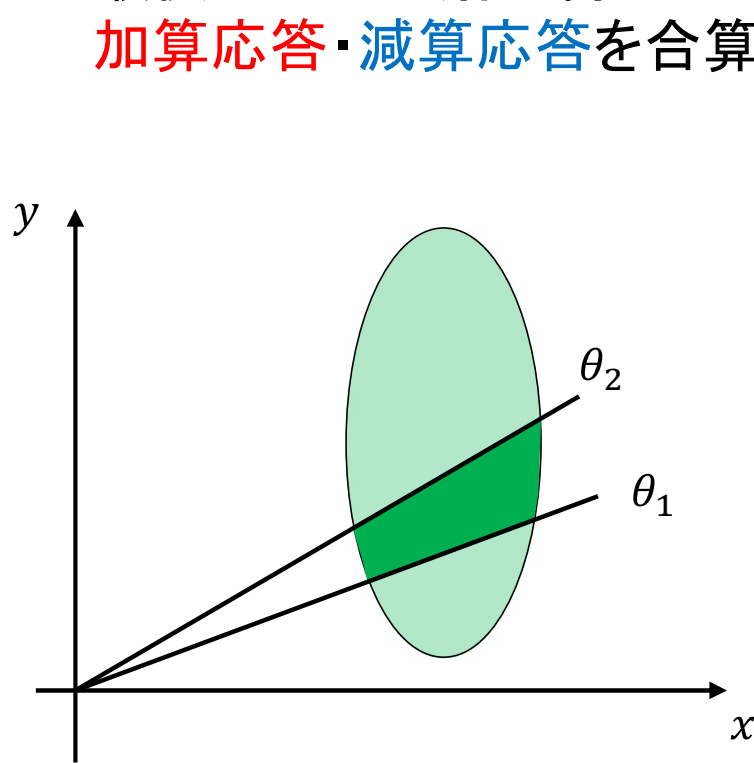
- コンクリート舗装の設計では「コンクリート版下面に発生する最大主応力」を用いるのが一般的である.
- Westergaard(1926)の研究による載荷公式が用いられることが多い.
- Pickett & Ray(1951)が, Westergaardの研究成果を基に, 複数の載荷重による応答を算出可能な影響図を作成した.
- 我が国では, 福手(1977)がPickett & Rayの研究成果を基にプログラム化し, 任意の位置・任意の方向の応力を算出可能とした.



Pickett & Rayが作成した影響図
(左はたわみ算出用・右はモーメント算出用)

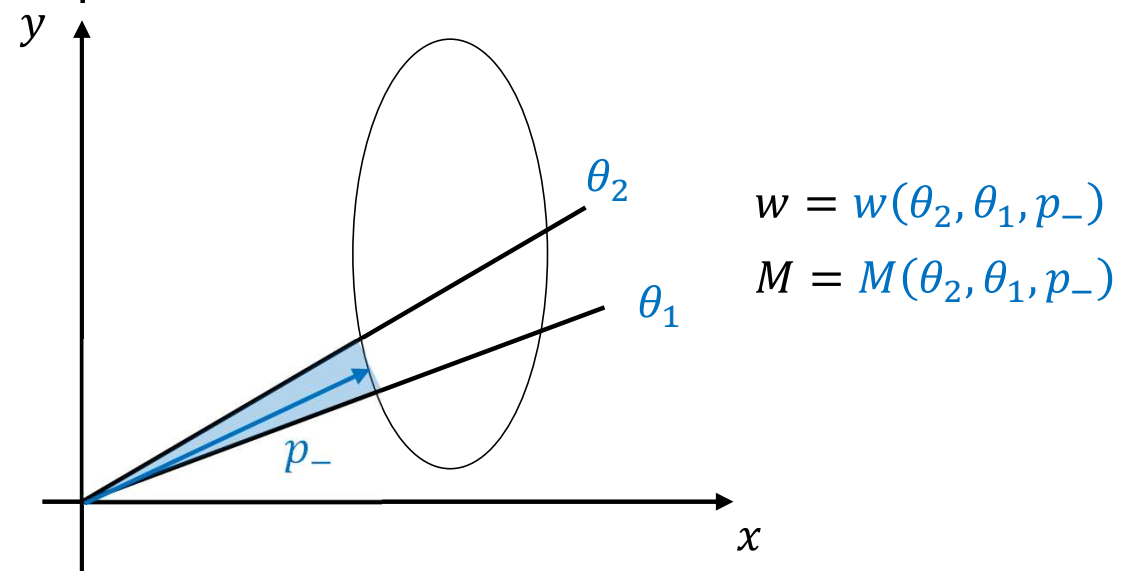
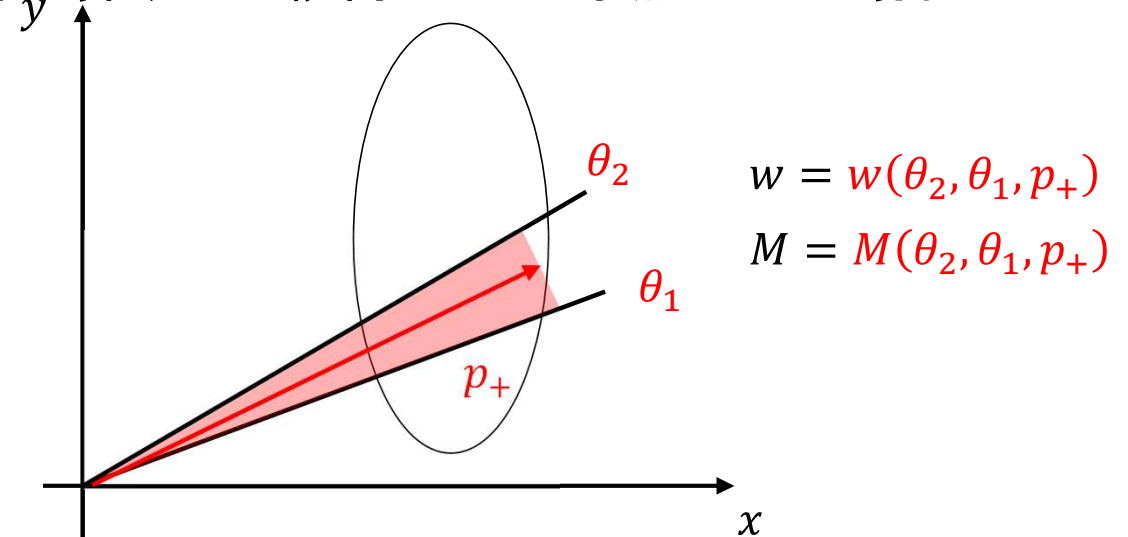
現プログラムの概要

- スキャン角度を細かく変えて、載荷重を小さな扇形荷重に分割する。
- 原点からの長さ p ・角度 θ の小さな扇形荷重の応答を, Pickett & Rayの研究成果を用いて算出する。
- 最後に全ての扇形荷重の応答を合算する。載荷重内に原点がない場合は
加算応答・減算応答を合算する。



$$w = w(\theta_2, \theta_1, p_+) - w(\theta_2, \theta_1, p_-)$$

$$M = M(\theta_2, \theta_1, p_+) - M(\theta_2, \theta_1, p_-)$$



【参考】 基本式

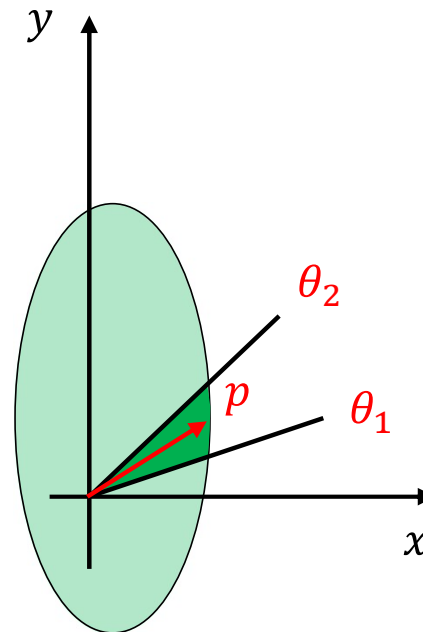
扇形荷重による原点のたわみ (Pickett & Ray, 1951)

$$w(\theta_2, \theta_1, p) = \frac{q}{K} \cdot \frac{\theta_2 - \theta_1}{2 \cdot \pi} \cdot \left\{ 1 + \frac{\pi}{2} \cdot \frac{p}{l} \cdot \text{Im} \left[\sqrt{i} \cdot H_1^1 \left(\sqrt{i} \cdot \frac{p}{l} \right) \right] \right\}$$

扇形荷重による原点のx方向モーメント (Pickett & Ray, 1951)

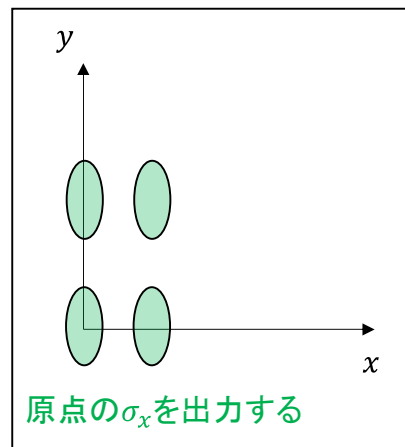
$$M(\theta_2, \theta_1, p) = \frac{q \cdot l^2}{8} \cdot \text{Re}[Z]$$

$$Z = (1 + \mu) \cdot (\theta_2 - \theta_1) \cdot \frac{p}{l} \cdot \sqrt{i} \cdot H_1^1 \left(\sqrt{i} \cdot \frac{p}{l} \right) \\ + (1 - \mu) \cdot (\sin 2\theta_2 - \sin 2\theta_1) \cdot \left\{ \sqrt{i} \cdot \frac{p}{2 \cdot l} \cdot H_1^1 \left(\sqrt{i} \cdot \frac{p}{l} \right) + H_0^1 \left(\sqrt{i} \cdot \frac{p}{l} \right) - 0.5 \right\}$$

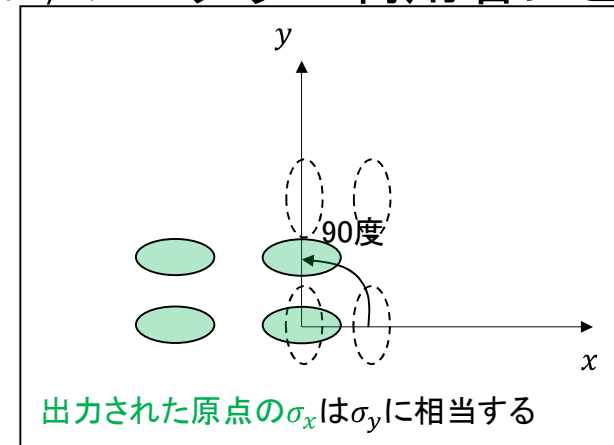


現プログラムの概要

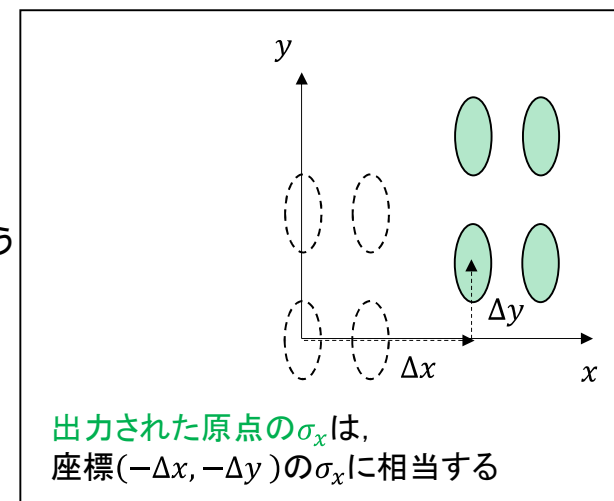
- 【**原点におけるたわみ**】【**原点におけるx方向応力**】を算出する.
- 脚荷重角度と脚荷重シフト量を入力することで, 任意の位置・方向の応力を算出できる. 特に, 最大主応力を確認する場合は0-179度まで小刻みに解析を行う必要がある.
- 【荷重を回転・シフトさせる】という概念が, プログラム利用者にとって若干わかりにくい.



σ_y を算出するためには
脚荷重を90度回転させる



応答出力点が原点となるよう
脚荷重をシフトさせる



改良方針

脚荷重角度と脚荷重シフト量

- 【荷重の角度を変える】【荷重の位置をシフトさせる】を意識しなくて済むようにする.
 - 出力座標を入力するだけでx方向応力・y方向応力と最大主応力を出力する.

荷重形状

- 楕円形のみを対象としていたため, 円形荷重, 矩形荷重にも対応させる.
 - 円形荷重対応は容易(単なる楕円形的一种).
 - 矩形荷重対応は荷重スキャン方法の考案する必要がある.

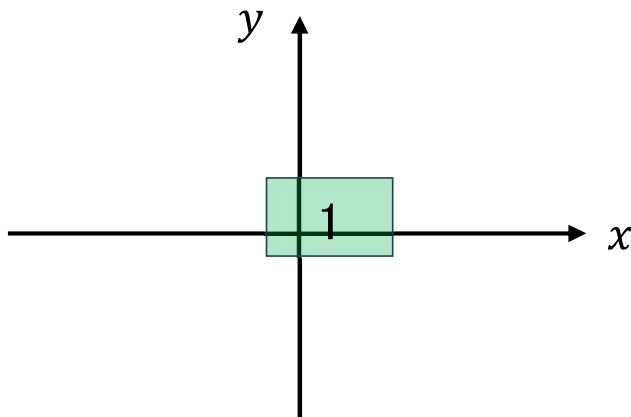
その他

- 複素数の引数に対するハンケル関数値が必要となる
 - 現プログラムではベッセル関数・変形ベッセル関数により算出しているが, コードを解読・理解できなかった.
 - VBAにはハンケル関数が用意されていない.
 - VBAにはベッセル関数はあるが複素数に対応していない.

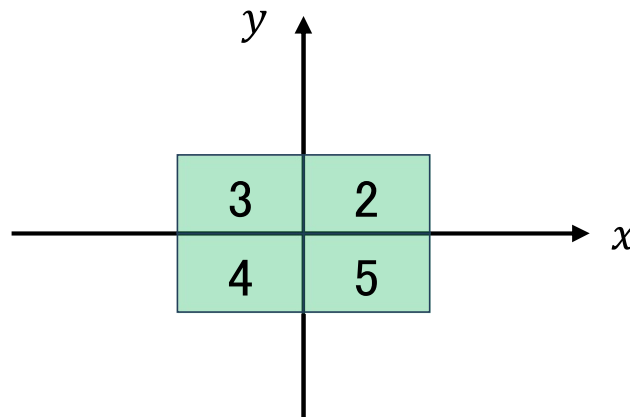
矩形荷重に対する荷重スキャン

- 原点, x 軸, y 軸と矩形荷重との位置関係を整理すると25パターン, さらに, それぞれについても数パターンに場合分けが必要.
- 力技で処理することもできなくはないが, 効率的な処理方法を検討した.

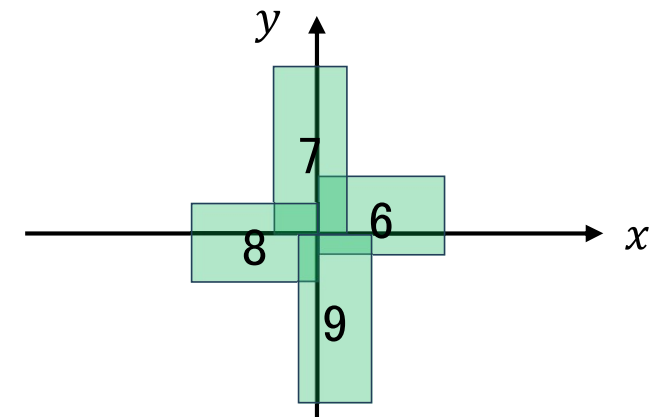
原点が矩形内



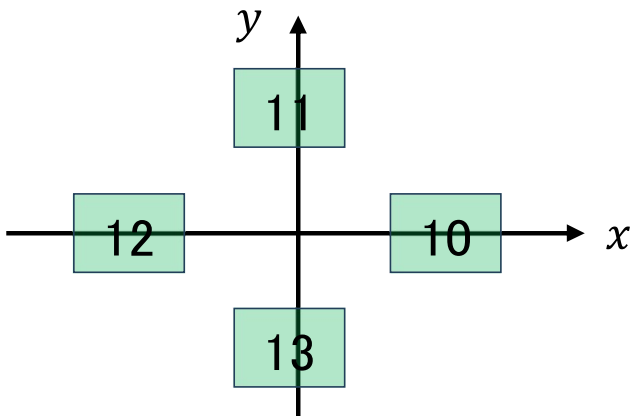
x 軸と y 軸の両方に接する



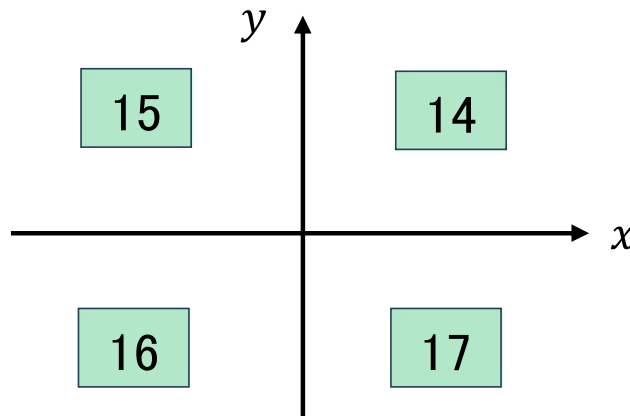
x 軸か y 軸に接し, 軸を跨ぐ



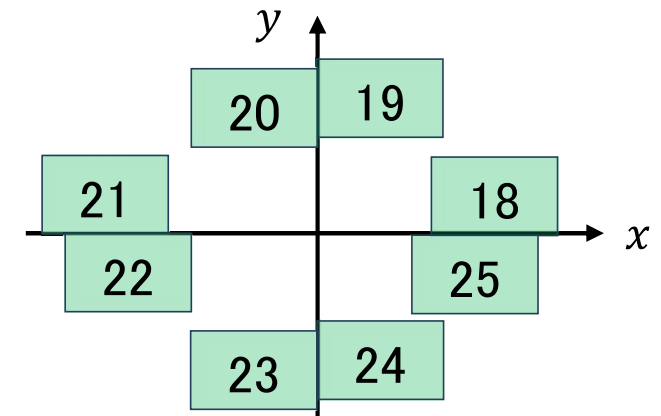
片方の軸を跨ぐ



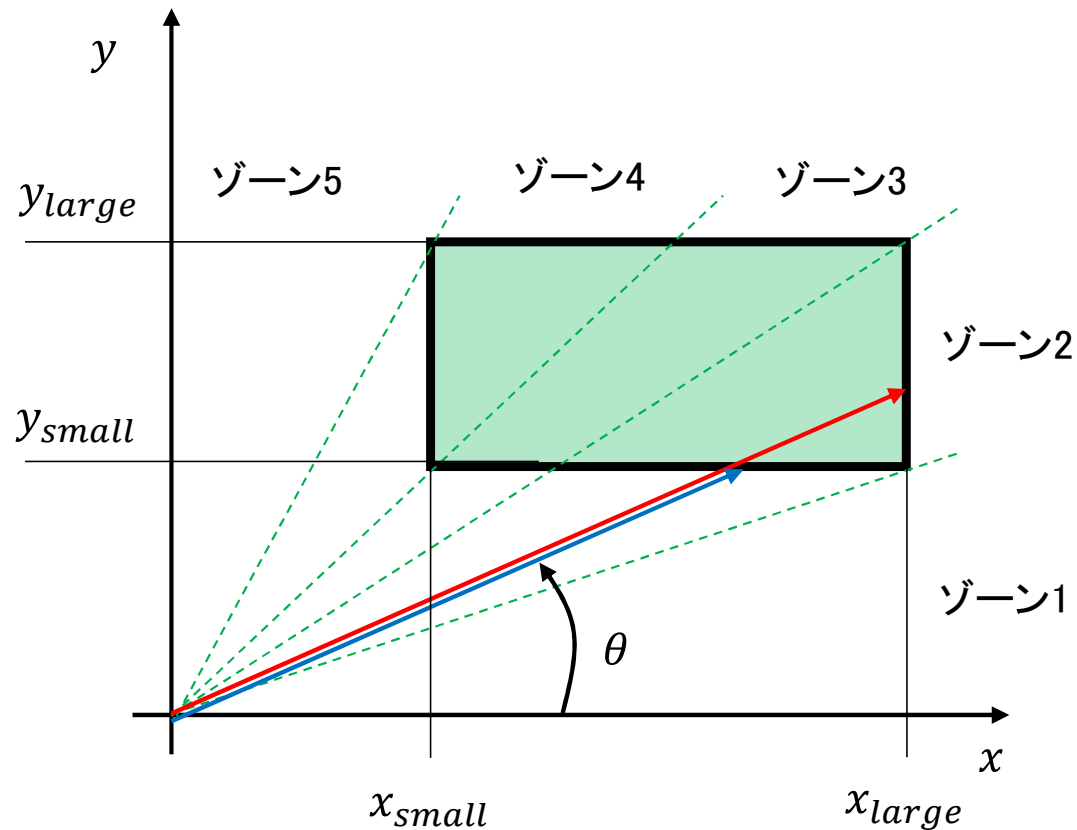
軸を跨がない



x 軸か y 軸に接し, 軸を跨がない



矩形荷重に対する荷重スキャン



ゾーン1
荷重スキャンは不要

ゾーン2
 p_+ は x_{large} で決定され, $x_{large}/\cos\theta$
 p_- は y_{small} で決定され, $y_{small}/\sin\theta$

ゾーン3
 p_+ は y_{large} で決定され, $y_{large}/\sin\theta$
 p_- は y_{small} で決定され, $y_{small}/\sin\theta$

ゾーン4
 p_+ は y_{large} で決定され, $y_{large}/\sin\theta$
 p_- は x_{small} で決定され, $x_{small}/\cos\theta$

ゾーン5
荷重スキャンは不要

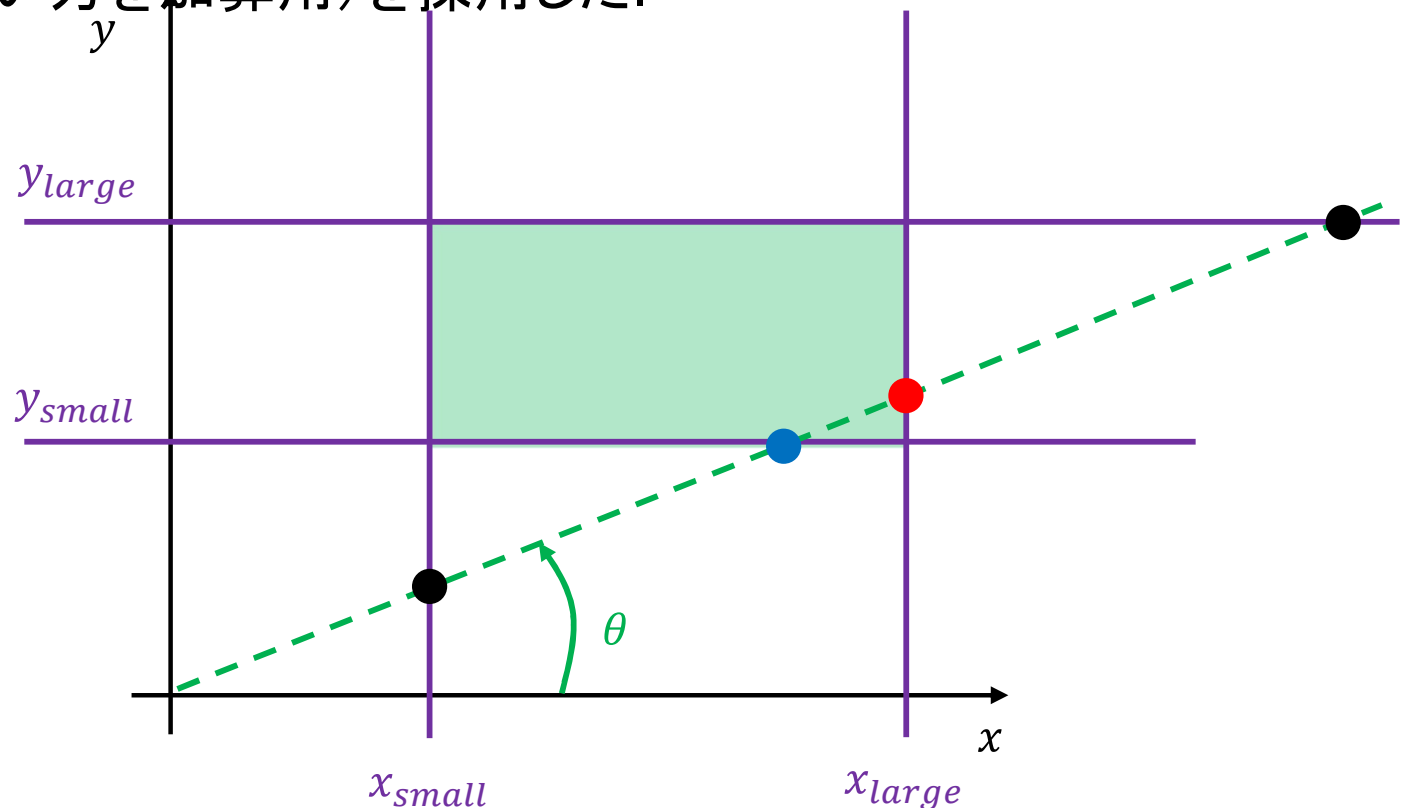
これを25パターン別にかき分ける??
(可能であるが避けたい)

矩形荷重に対する荷重スキャン

いろいろと試行錯誤した結果

- 【スキャン角度 θ のスキャン直線】と【矩形を構成する4つの直線】の交点は必ず4つ存在する. 原点から交点までの距離は p .
- 4つの p は正・0・負となるが, 必要なのは【短い方から二番目と三番目の p (ただし正のみ採用)】であることがわかった.
- 円形荷重・楕円形荷重も同様の荷重スキャン方法 (正の2つの p のうち短い方を減算用・長い方を加算用) を採用した.

- 青点は減算応答値用の交点
- 赤点は加算応答値用の交点
- 黒点は採用しない交点



ハンケル関数の処理方法

- VBAにはハンケル関数が用意されていない。ベッセル関数はあるが複素数の引数に対応していない。
- プログラム内で計算することはせずに、多数のハンケル関数の真値をEXCELシートに格納し、この真値から近似的に算出することとした。

$$\alpha = \frac{p}{\sqrt{2} \cdot l} \quad \text{とすれば}$$

$$\text{第1種1次} \quad H_1^1\left(\sqrt{i} \cdot \frac{p}{l}\right) = H_1^1(\alpha + \alpha \cdot i) = A_1(\alpha) + B_1(\alpha) \cdot i$$

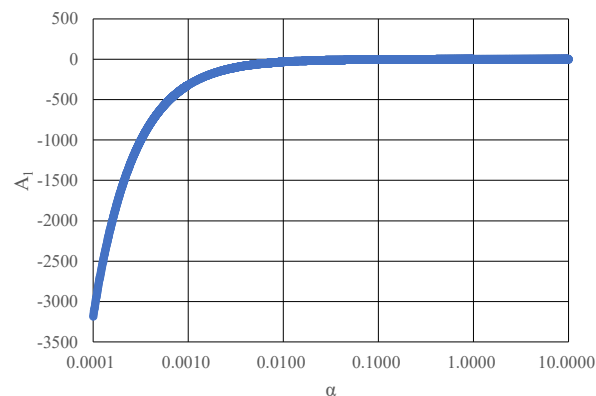
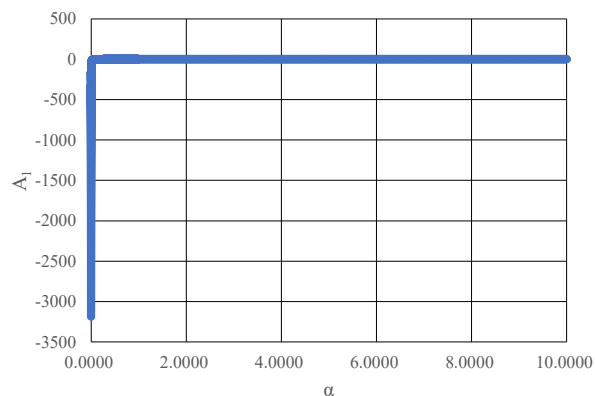
$$\text{第1種0次} \quad H_0^1\left(\sqrt{i} \cdot \frac{p}{l}\right) = H_0^1(\alpha + \alpha \cdot i) = A_0(\alpha) + B_0(\alpha) \cdot i$$

であり、計算に必要なのは $A_1(\alpha), B_1(\alpha), A_0(\alpha)$ の3つの実数。

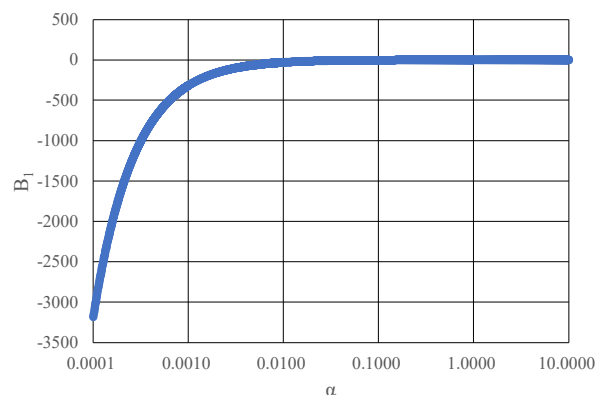
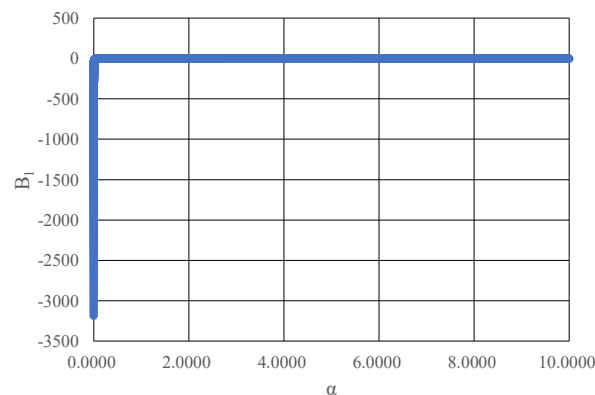
- ハンケル関数の真値は、カシオ計算機株式会社のWEBサイト「keisan サービス」で連続解析により算出した。
- 引数 α の増分を小さくして大量の真値を格納すれば精度は向上するが、どの程度の真値を用意すべきかを検討した。

ハンケル関数の処理方法

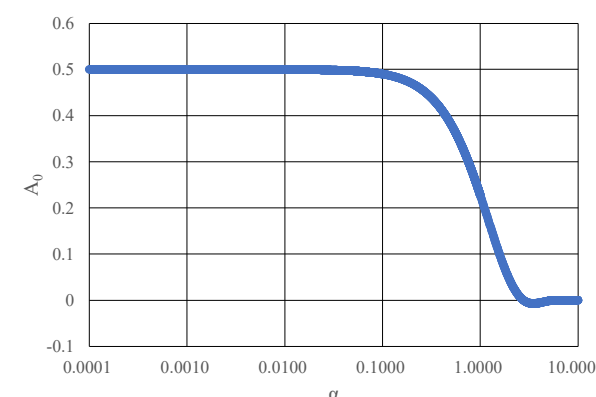
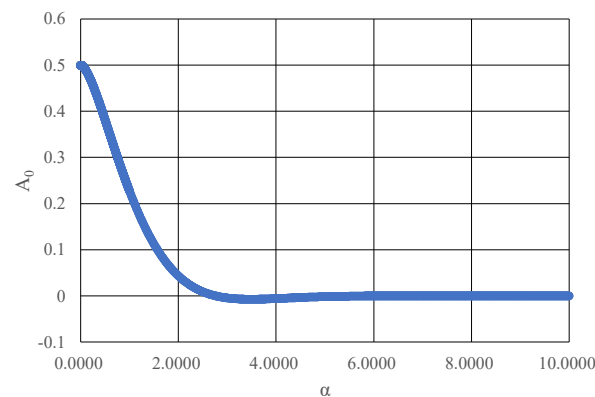
A1



B1



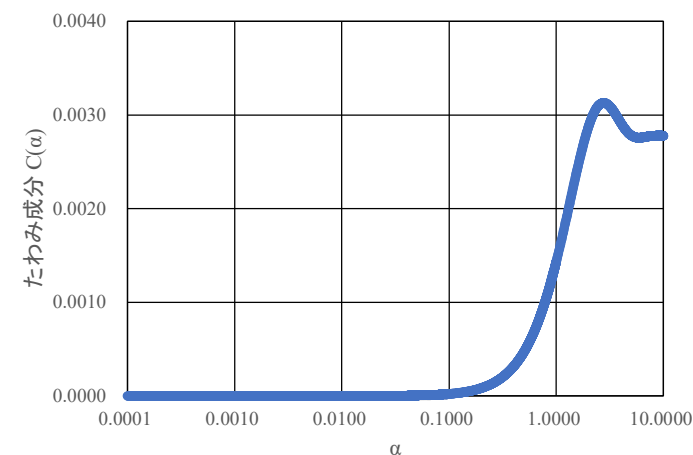
A0



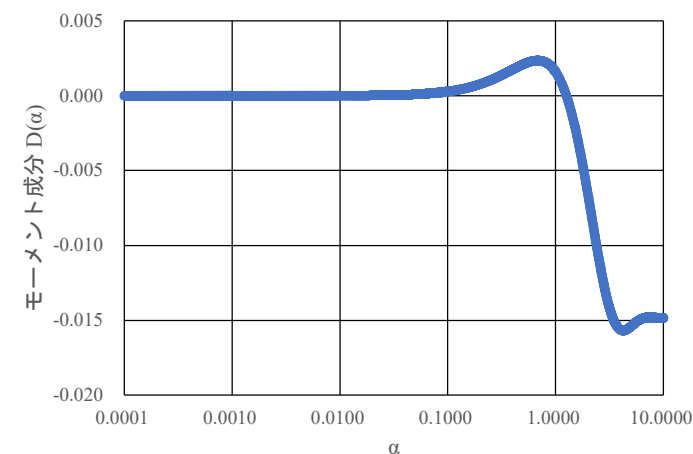
横軸は通常軸

横軸は対数軸

たわみ成分



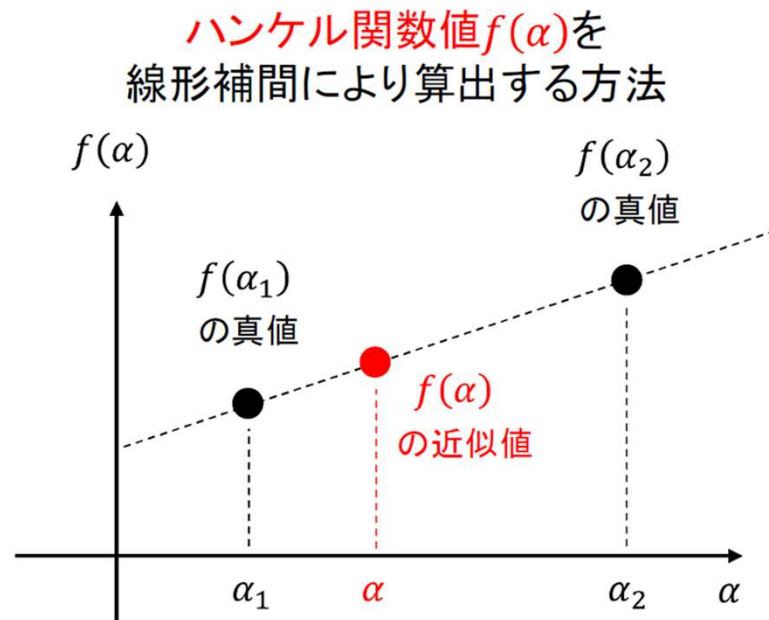
モーメント成分



横軸は対数軸

ハンケル関数の処理方法

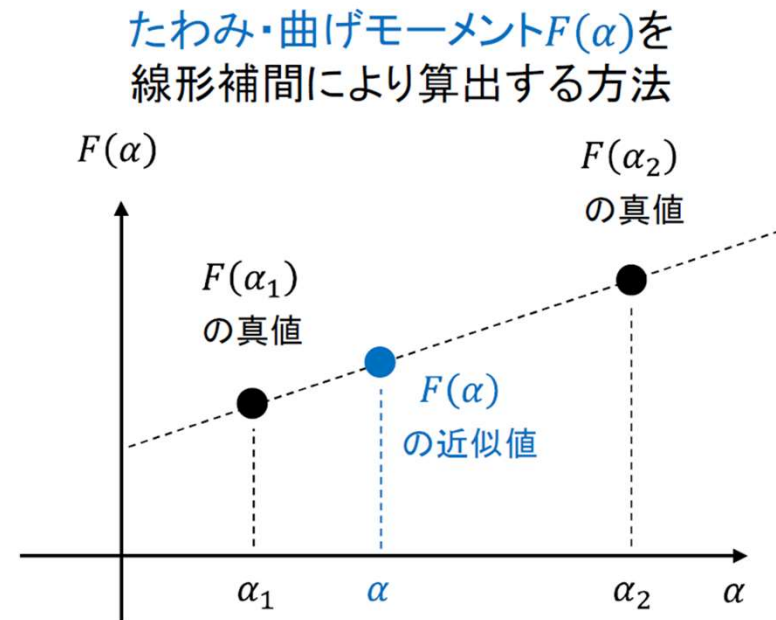
- 線形補間により **ハンケル関数値** を精度よく算出するのであれば、一部の領域の α において多数の真値が必要となる(左図)
- 線形補間により **たわみ・曲げモーメント** を精度よく算出するのであれば、少ない真値でも問題ない(右図).



$$f(\alpha) = \frac{f(\alpha_2) - f(\alpha_1)}{\alpha_2 - \alpha_1} \cdot (\alpha - \alpha_1) + f(\alpha_1)$$



ハンケル関数値 $f(\alpha)$ により
たわみ・曲げモーメント $F(\alpha)$ を算出



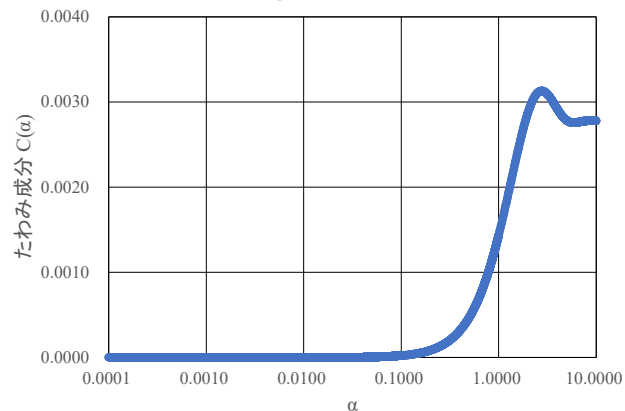
$$F(\alpha) = \frac{F(\alpha_2) - F(\alpha_1)}{\alpha_2 - \alpha_1} \cdot (\alpha - \alpha_1) + F(\alpha_1)$$

ハンケル関数の処理方法

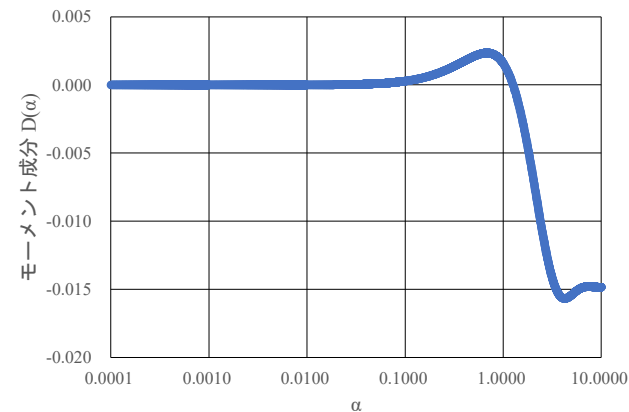
- 4501通りのハンケル関数値をプログラム内に格納することとした.
- 格納した4501通りの中から α の前後の $\alpha 1$ と $\alpha 2$ を特定→ $\alpha 1$ と $\alpha 2$ の応答値を算出→ α の応答値を線形補間で算出.

α の範囲	α の間隔	α の数	最大誤差(%)	
			たわみ成分	モーメント成分
0.0001～0.001	0.000001	900	0.002	0.002
0.001～0.01	0.00001	900	0.002	0.002
0.01～0.1	0.0001	900	0.002	0.002
0.1～1	0.001	900	0.002	0.001
1～10	0.01	901	0.001	0.403

たわみ成分

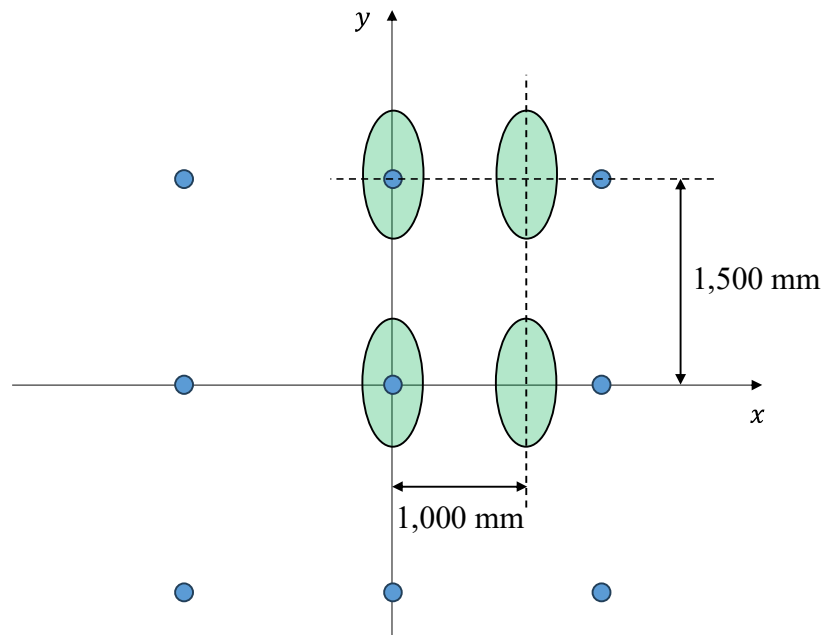


モーメント成分



精度の確認

現プログラムの解析結果とは概ね0.35%程度の差であった。ただし【現プログラムが有効数字三桁までしか出力できない】ことに起因する差, 【現プログラムよりもスキャン角度幅が小さい】ことに起因する差を含んでいる。



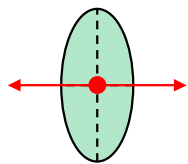
x	y	たわみの差 (%)	最大主応力の差 (%)
0	0	0.219	0.311
1500	0	0.281	0.321
1500	1500	0.281	0.321
0	1500	0.219	0.311
-1500	1500	0.484	0.457
-1500	0	0.484	0.457
-1500	-1500	0.173	0.506
0	-1500	0.400	0.374
1500	-1500	0.281	0.428

最大主応力の発生位置及び方向の傾向

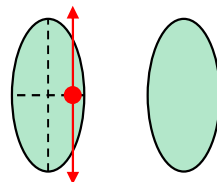
- 1脚2輪 載荷重中心からやや内側・縦断方向.
横中心間隔の影響が比較的大きい.
- 1脚4輪 載荷重中心からやや内側・概ね30度程度
間隔の影響は大きくはない.
- 1脚6輪 縦断方向中間の載荷重中心からやや内側・横断方向.
間隔の影響は大きくはない.

最大主応力を算出する際の参考情報(どのあたりの応力を出すべきか)として、以下の解析結果を示すこととした.

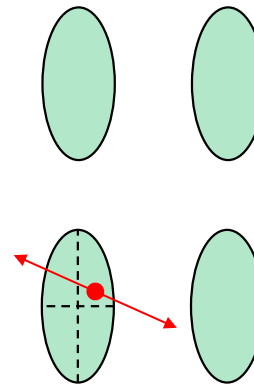
1脚1輪



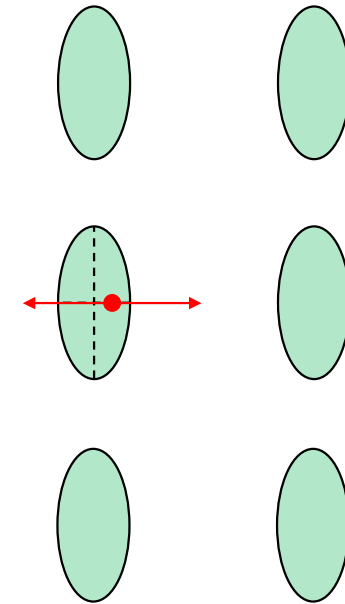
1脚2輪



1脚4輪



1脚6輪



結論

- ① 矩形荷重・円形荷重・楕円形荷重に対する効率的な荷重スキャン方法を整理した.
- ② ハンケル関数がたわみ成分・モーメント成分に及ぼす誤差の傾向を明らかにし, 効率的なハンケル関数処理方法を整理した.
- ③ 最大主応力の発生位置及び方向の傾向を明らかにした.

プログラムの使用については
国総研・空港施設研究室HPをご覧ください

<https://www.ysk.nilim.go.jp/kakubu/kukou/sisetu/index.html>