C. Δ-form, Define residual. Consider $[\boldsymbol{\delta}^2{}_x + \boldsymbol{\delta}^2{}_y]\,\boldsymbol{\psi} = \mathbf{A}\cdot\boldsymbol{\psi}$

Concept,h. Let $\boldsymbol{\psi} = \boldsymbol{\psi}^n + \Delta\boldsymbol{\psi}$

$$\mathbf{A}\cdot\Delta\boldsymbol{\psi} = \boldsymbol{\omega} - \mathbf{A}\cdot\boldsymbol{\psi}^n \equiv \mathbf{R}$$

Note that $\mathbf{R} = \mathbf{0}$ if the equation is satisfied (could search for $\|\mathbf{R}\|=0$). Algorithm is

$$\Delta\boldsymbol{\psi} = \mathbf{A}^{-1}\cdot\mathbf{R}\,;\ \boldsymbol{\psi}^{n+1} = \boldsymbol{\psi}^n + \Delta\boldsymbol{\psi}$$

NB: solution when $|\Delta\boldsymbol{\psi}|$ becomes small. Then

$|\mathbf{R}| \to 0$. Convergence of residual; check, $\mathbf{R} = \mathbf{0}$ is the original equation. (comment, change $\mathbf{A}$)

GS: Write scheme as $A_1\psi_1 + A_2\psi_2 + A_3\psi_3 + A_4\psi_4 + A_5\psi_5 = \omega$

Let $\psi_3^{n+1} = \psi_3 + \Delta\psi_3$ (that is $\psi_{i,j} = \psi_3$)

$$A_3\Delta\psi_3 + A_1\psi_1 + A_2\psi_2 + A_3\psi_3 + A_4\psi_4 + A_5\psi_5 = \omega$$

Define residual by (sweep LL to UR) *but, superscripts don't exist*

$$R = \omega-(A_1\psi_1^{n+1} + A_2\psi_2^{n+1} + A_3\psi_3^n + A_4\psi_4^n + A_5\psi_5^n)$$

$$\Delta\psi = R/A_3 \text{ and } \psi_{i,j}^{n+1} = \psi_{i,j}^n + \Delta\psi$$

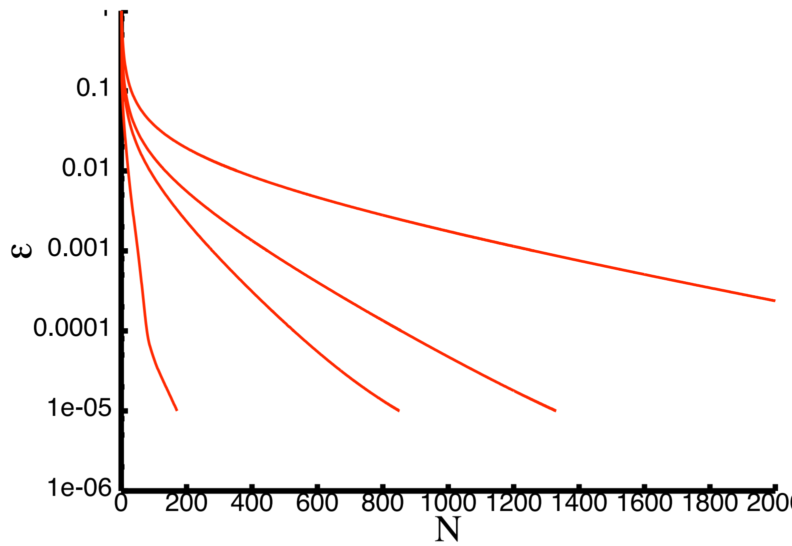
Then Gauss-Seidel scheme is (in pseudo-code)

```
Initialize ψ including boundary values
WHILE (ε > 10^-5)
DO j = 2,J-1
  DO i = 2,I-1
     R = ω (i,j)-(A1ψ (i,j-1) + A2ψ (i-1,j) + A3ψ (i,j) + A4ψ (i+1,j) + A5ψ (i,j+1))
     Δψ    = R/A3
     ψ(i,j) = ψ(i,j) + Δψ
     ε     = ε + Δψ^2    or  ε = ε + |Δψ|   or  ε = max(ε, |Δψ|)
  ENDDO  i
ENDDO  j
     IF (ε0 = 0)  ε0 = ε
     ε    = sqrt(ε/ε0)        : rms error
ENDWHILE
```

Error measures: L_2 = r.m.s., L_1 = avg. abs, $L_n = (\Sigma\ |\Delta\psi|^n)^{1/n}$, $L_\infty = \max_{i,j}(|\Delta\psi|)$.

Convergence $\Delta\psi\to 0$ corresponds to $R\to 0$. So iteration is driving residual to 0.
Computational complexity from do loops: #ops ~ J*I x # iterations = N x # iterations

† Comment: Let $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$. Then $\mathbf{R} = \boldsymbol{\omega} - \mathbf{L}\psi^{n+1} + \mathbf{D}\psi^n + \mathbf{U}\psi^n$ and $\Delta\boldsymbol{\psi} = \mathbf{D}^{-1}\,\mathbf{R}$



Why not sweep UR to LL? Fine:

```
DO WHILE (ε > 10^-5)
DO j= J-1,2,-1
 DO i=I-1,2,-1
    ............
    Δψ     = R(i,j)/A3
    ψ(i,j)  = ψ(i,j) + Δψ
    ............
 ENDDO   i
ENDDO   j
  ..........
ENDWHILE
```

$$R = \omega-(A_1\psi_1^n + A_2\psi_2^n + A_3\psi_3^n + A_4\psi_4^{n+1} + A_5\psi_5^{n+1})$$

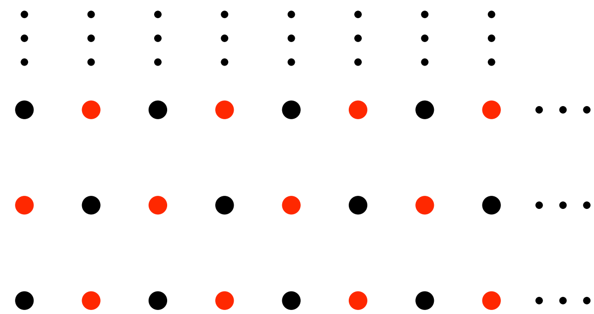
How about both? Symmetric G-S: sweep LL to UR then UR to LL: just combine above.

```
DO j = 2,J-1
 DO i = 2,I-1
 ..........
 ENDDO   i
ENDDO   j
DO j= J-1,2,-1
 DO i=I-1,2,-1
  ........
 ENDDO   i
ENDDO   j
```

Another variant is red-black ordering (vectorizing):

```
DO j= 2,J-1
 ist = 2+ [j-(j/2)*2]  ! (2+j_mod 2)
 DO i=ist,I-1,2
  ------
     Δψ     = R(i,j)/A3
   ψ(i,j)  = ψ(i,j) + Δψ
  ------
ENDDO   i
ENDDO   j
```

$R = \omega\text{-}(A_1\psi_1^n + A_2\psi_2^n + A_3\psi_3^n + A_4\psi_4^n + A_5\psi_5^n)$



Then update red points

```
DO j= 2,J-1
 ist = 3 - [j-(j/2)*2]  ! (3-j_mod 2)
 DO i=ist,I-1,2
  ------
     Δψ     = R(i,j)/A3
   ψ(i,j)  = ψ(i,j) + Δψ
  ------
 ENDDO   i
ENDDO   j
```

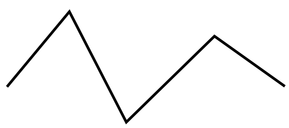
$R = \omega\text{-}(A_1\psi_1^{n+1} + A_2\psi_2^{n+1} + A_3\psi_3^{n+1} + A_4\psi_4^{n+1} + A_5\psi_5^{n+1})$

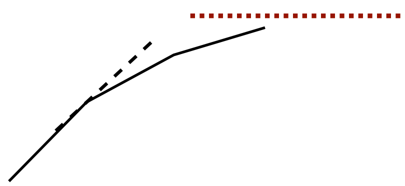
Does GS converge? Yes: diagonal dominance, $|A_3| = A_1 + A_2 + A_4 + A_5 = (2/\Delta x^2 + 2/\Delta y^2)$

Can convergence be accelerated? Yes, several methods

Acceleration of convergence: Simplest method: Successive over relaxation (SOR).
Recall $\mathbf{R} = \boldsymbol{\omega} - \mathbf{A} \cdot \boldsymbol{\psi}^n$; $\Delta\psi = R/A_3$. Computational complexity: N x number of iterations. Reduced from N^2; now want to reduce # iterations.

How to relax? Might try averaging $(\psi^{n+1} + \psi^n)/2 = \psi^n + \Delta\psi/2 = \psi^{n+1}$. Actually slows G-S down.





Intuitively: if oscillating under relax -- averaging smooths error; if extrapolating over relax

G-S with over relaxation = SOR:

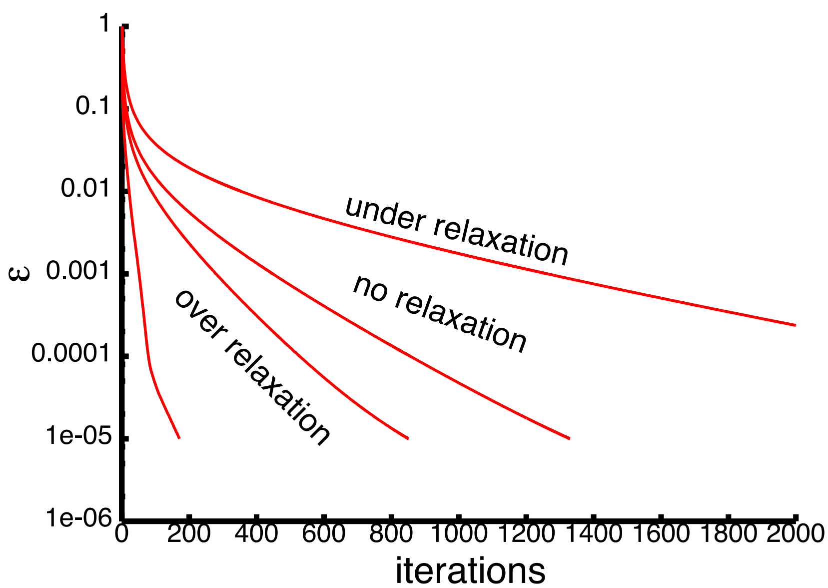
$A_3\Delta\psi_3 + A_1\psi_1 + A_2\psi_2 + A_3\psi_3 + A_4\psi_4 + A_5\psi_5 = \omega$

$R = \omega - (A_1\psi_1^{n+1} + A_2\psi_2^{n+1} + A_3\psi_3^n + A_4\psi_4^n + A_5\psi_5^n)$

$\Delta\psi = R/A_3$. Now let $\psi_{i,j}^{n+1} = \psi_{i,j}^n + \lambda\Delta\psi$;

Alternatively if $\psi_{G\text{-}S} = \psi_{i,j}^n + \Delta\psi$ then $\psi_{i,j}^{n+1} = (1-\lambda)\psi_{i,j}^n + \lambda\, \psi_{G\text{-}S}$ i.e., average of G-S and old value --- except that $\lambda > 1$!

$\lambda = 1$ G-S
$\lambda > 1$ over relaxation
$\lambda < 1$ under relaxation



CFD (SIMPLE) under relaxation is used. But for G-S over relaxation is needed.

Can prove $\lambda < 0$ or $\lambda > 2$ is unstable. Optimal λ is usually empirical -- depends on eigenvalues of **A**. For Laplace equation on uniform grid, can show optimal is $1 < \lambda < 2$.

D. Pseudo-code for SOR

```
GS: WHILE(err > errmx)
   err  = 0
   DO j=2,J-1
     DO i=2,I-1
!------------
! evaluate residual ; Compute Delta's
!------------
       R = omeg(i,j) -A(1)*ψ(i,j-1) -A(2)*ψ(i-1,j)
                     -A(4)*ψ(i+1,j) -A(5)*ψ(i,j+1)-A(3)*ψ(i,j)
       Δψ(i,j)=  R/A(3)
       ψ(i,j) = ψ(i,j) + λ Δψ
       err    = err + |Δψ|
     ENDDO i
   ENDDO j
   err = err/err0
  ENDWHILE GS
```

E. comments: Linear algebra (theory)
(**NB**: Not L, U of L-U decomposition!!)

$\mathbf{A} \cdot \psi = \boldsymbol{\omega}$

$(\mathbf{L}+\mathbf{D}+\mathbf{U}) \cdot \boldsymbol{\psi} = \boldsymbol{\omega}$; G-S is $(\mathbf{L}+\mathbf{D}) \cdot \boldsymbol{\psi}^{n+1} = \boldsymbol{\omega} - \mathbf{U} \cdot \boldsymbol{\psi}^{n}$ (like back substitution, without first doing Gauss elimination)

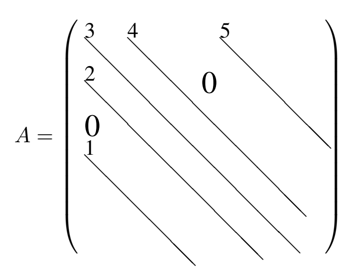
Error $\boldsymbol{\varepsilon}^n = \boldsymbol{\psi}^n - \boldsymbol{\psi}^{exact}$

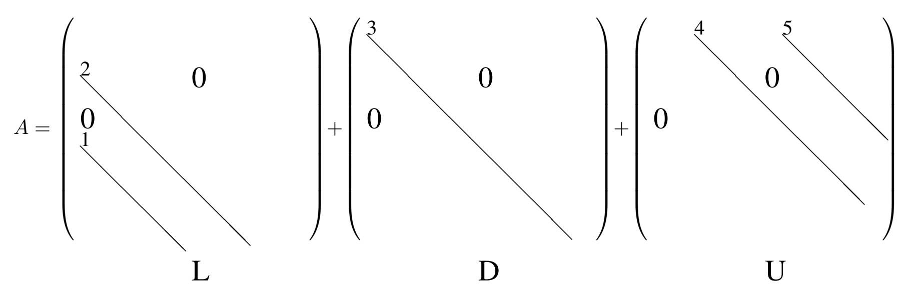
$(\mathbf{L}+\mathbf{D}) \cdot \boldsymbol{\psi}^{exact} = \boldsymbol{\omega} - \mathbf{U} \cdot \boldsymbol{\psi}^{exact}$

$(\mathbf{L}+\mathbf{D}) \cdot \boldsymbol{\varepsilon}^{n+1} = -\mathbf{U} \cdot \boldsymbol{\varepsilon}^{n}$; $\boldsymbol{\varepsilon}^{n+1} = -(\mathbf{L}+\mathbf{D})^{-1} \cdot \mathbf{U} \cdot \boldsymbol{\varepsilon}^{n} \equiv \mathbf{M} \cdot \boldsymbol{\varepsilon}^{n}$.

M is *iteration* matrix. $\boldsymbol{\varepsilon}^n = \mathbf{M}^{n-1} \cdot \boldsymbol{\varepsilon}^1 \rightarrow 0$.

Converges if eigenvalues of **M** are less than unity.





Theorem (diagonal dominance)

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \dots \\ a_{21} & a_{22} & a_{23} \dots \\ a_{31} & a_{32} & a_{33} \dots \\ \dots & \dots & \dots \end{bmatrix}$$

If

$$|a_{ii}| \geq \sum_{j=1,\ j\neq i}^{J} |a_{ij}|$$

for all i with inequality for at least one, then G-S converges. Rule of thumb: increase diagonal dominance to improve convergence.

For Laplace on uniform grid

$$\left|\frac{2}{\Delta x^2} + \frac{2}{\Delta y^2}\right| = \left|\frac{1}{\Delta x^2}\right| + \left|\frac{1}{\Delta x^2}\right| + \left|\frac{1}{\Delta y^2}\right| + \left|\frac{1}{\Delta y^2}\right|$$

Artificial time-stepping adds $1/\Delta\tau$ to diagonal.

$\nabla^2\psi = \partial\psi/\partial\tau \rightarrow \mathbf{A} \cdot \Delta\psi = \mathbf{R} + \Delta\psi/\Delta\tau \rightarrow [\mathbf{A}-\mathbf{I}/\Delta\tau]\ \Delta\psi = R$; diagonal becomes

$$-\frac{2}{\Delta x^2} - \frac{2}{\Delta y^2} - \frac{1}{\Delta\tau}$$

Background to other methods: framework is drive residual to zero in $A\Delta\psi = R$

$$\nabla^2\psi = [\delta^2_x + \delta^2_y]\,\psi = A \cdot \psi = \omega; \quad \text{G-S} \quad \text{Let } \hat{\mathbf{A}} \equiv (\mathbf{L+D})$$

$$\hat{A} \cdot \psi^{n+1} = \omega - U \cdot \psi^n \;\Rightarrow\; \hat{A} \cdot \Delta\psi^{n+1} = U \cdot \Delta\psi^n$$

Note also

$$\hat{\mathbf{A}} \cdot \Delta\psi^{n+1} = -\mathbf{U} \cdot \Delta\psi^n = -A_4 \cdot \Delta\psi_4{}^n - A_5 \cdot \Delta\psi_5{}^n$$

Start with $\psi^0 = 0$. Then $\Delta\psi^1 = \psi^1 = (\mathbf{L}+\mathbf{D})^{-1}\,\omega$ --- i.e., one Gauss-Seidel sweep.

F. Other acceleration methods:

Multi-grid -- smooth error on coarse grids for efficiency. (See Pletcher et al).
Krylov space (Conjugate-Gradient, BiCG, GMRES) -- Search in solution space.