Iterační metody - metoda GMRES

1. Stáhněte si soubory `GMRES.m` a `test.m` z `homel.vsb.cz/~dom0015/IM/kody/07_GMRES/` a uložte je do vhodného adresáře.

2. Vyřešte pomocí GMRES soustavu $Ax = b$, kde $A \in \mathbb{R}^{100\times 100}$ je náhodná regulární matice a $b = \overline{1}$.

3. Porovnejte výsledek po $i \in \{10, 50, 90, 99, 100\}$ iteracích s matlabovským řešičem $A\backslash b$.

4. Vykreslete normu rezidua po $i = \{1, 2, \ldots, 100\}$ iteracích.

5. Upravte funkci GMRES.m tak, aby vracela matice V a H, co je v těchto maticích uloženo?

6. Ověřte, že platí vztah $AV_k = V_{k+1}H_{k+1,k}$.

7. Porovnejte normu rezidua po $i = \{1, 2, \ldots, 100\}$ iteracích s matlabovským řešičem `gmres(A,b,...)`.

8. Aplikujte funkci GMRES.m na soustavu se symetrickou maticí. Zdůvodněte výsledný tvar matice H.

9. Řešte soustavu se symetrickou maticí $A \in \mathbb{R}^{1000\times 1000}$ pomocí restartované metody gmres (použijte matlabovský řešič). Proveďte vždy 1000 iterací, nejprve s restartem po 250 iteracích, dále po 500 iteracích a nakonec bez restartu. Vykreslujte resvec(2:end), lepší bude asi použít logaritmické měřítko.

10. Porovnejte také výpočetní čas všech tří variant z bodu 9.

11. Pokuste se vyvolat "lucky break down".

Předpodmínění

1. Modifikujte kód CG(A,b,u,epsilon,maxit) na PCG(A,B,b,u,epsilon,maxit).
2. Řešte soustavu s náhodnou SPD maticí $A \in \mathbb{R}^{100\times 100}$ pomocí metody CG, zaznamenejte výpočetní čas a vektor průběžných norem rezidua.
3. Řešte tutéž soustavu pomocí PCG s předpodmíněním maticí $B = diag(diag(A))$, porovnejte rezidua a výpočetní čas.
4. Řešte tutéž soustavu pomocí PCG s Gauss-Seidelovým předpodmíněním, porovnejte rezidua a výpočetní čas.
5. Řešte tutéž soustavu pomocí PCG se symetrickým Gauss-Seidelovým předpodmíněním, porovnejte rezidua a výpočetní čas.
6. Řešte tutéž soustavu pomocí PCG s předpodmíněním pomocí neúplného LU rozkladu (např. `ilu(sparse(A),struct('type','ilutp','droptol',0.1))`), porovnejte rezidua a výpočetní čas.
7. Řešte tutéž soustavu pomocí PCG s předpodmíněním pomocí neúplného Choleského rozkladu (např. `ichol(sparse(A),struct('type','ict','droptol',1e-2))`), porovnejte rezidua a výpočetní čas.
8. Pro každý z předpodmiňovačů (body 3. - 7.) porovnejte spektrum původní matice A se spektrem předpodmíněné matice.