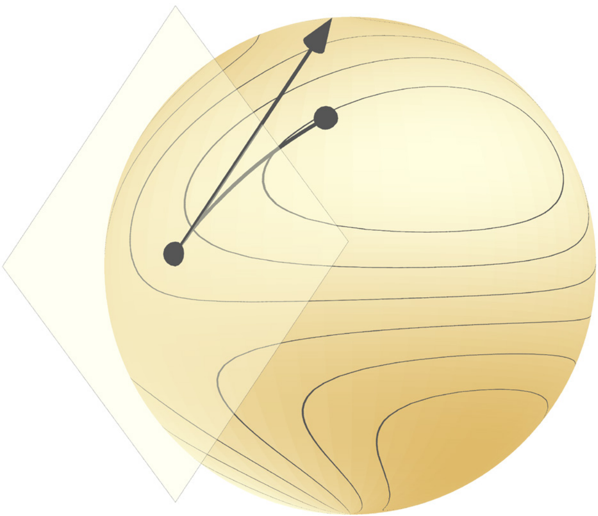
211

# Truncated conjugate gradients for the trust-region subproblem



Spring 2023

Optimization on manifolds, MATH 512 @ EPFL

Instructor: Nicolas Boumal

# Recall the trust-region subproblem (TRS)

$$f\big(R_x(v)\big) \approx m_x(v) \stackrel{\text{def}}{=} f(x) + \langle \operatorname{grad} f(x), v \rangle_x + \frac{1}{2}\langle v, \operatorname{Hess} f(x)[v] \rangle_x$$

We want an approximate solution of:

$$\min_{v \in \mathrm{T}_x\mathcal{M}} m_x(v) \quad \text{subject to} \quad \|v\|_x \leq \Delta$$

$\operatorname{Hess} f(x) \nsucceq 0$?

$\|v\|_x \leq \Delta$ ?

Strategy: run conjugate gradients, cautiously and opportunistically.

stop early.

$b = -\mathrm{grad} f(x)$

Truncated CG for $g(v) = \frac{1}{2}\langle v, Hv\rangle_x - \langle b, v\rangle_x$ on $\mathrm{T}_x\mathcal{M}$ with radius $\Delta$:

**Initialize** $v_0 = 0, r_0 = b, p_0 = r_0$

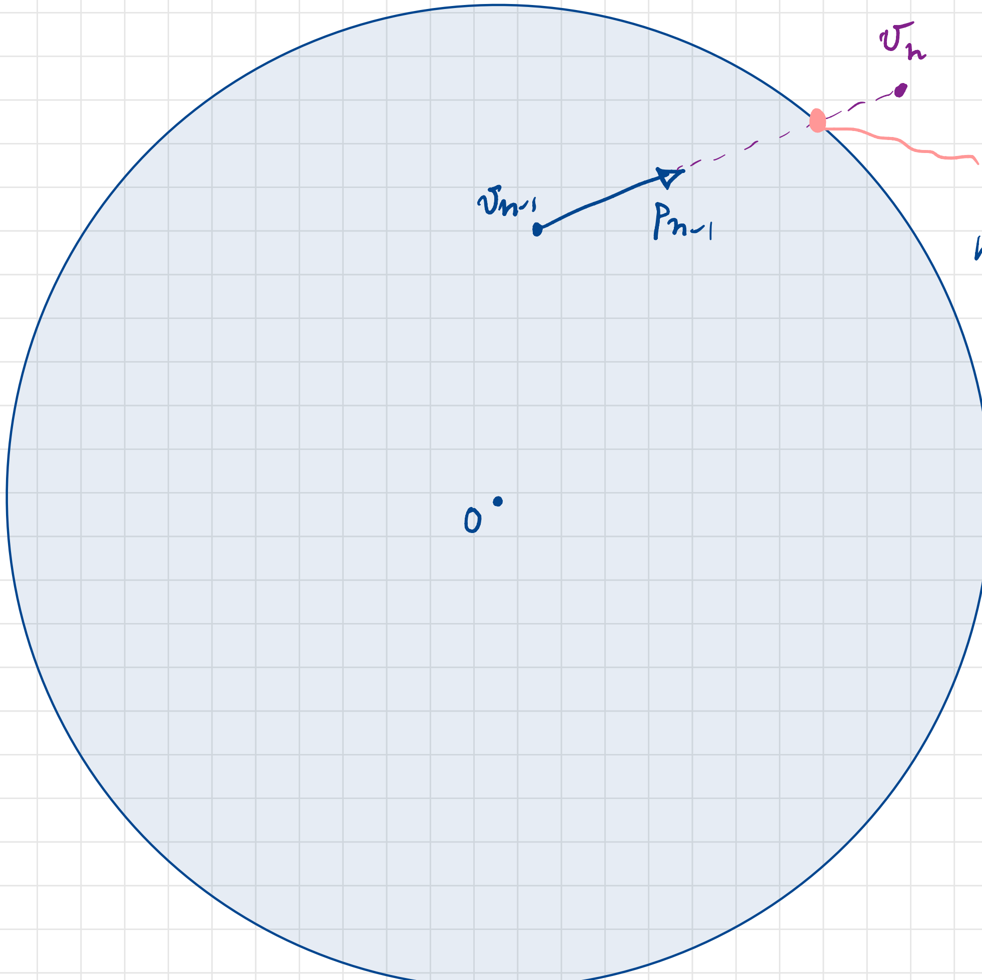
**For** $n$ in 1, 2, 3, …

- Compute $Hp_{n-1}$ and $\langle p_{n-1}, Hp_{n-1}\rangle_x$
- $\alpha_n = \frac{\|r_{n-1}\|_x^2}{\langle p_{n-1}, Hp_{n-1}\rangle_x}$
- $v_n = v_{n-1} + \alpha_n p_{n-1}$
- If $\langle p_{n-1}, Hp_{n-1}\rangle_x \leq 0$ or $\|v_n\|_x \geq \Delta$, then output $v_{n-1} + tp_{n-1}$ with $t \geq 0$ s.t. $\|v_{n-1} + tp_{n-1}\|_x = \Delta$
- $r_n = r_{n-1} - \alpha_n Hp_{n-1}$
- **If** $\|r_n\|_x \leq \min(0.1; \|b\|_x) \cdot \|b\|_x$, **output** $v_n$
- $\beta_n = \frac{\|r_n\|_x^2}{\|r_{n-1}\|_x^2}$
- $p_n = r_n + \beta_n p_{n-1}$

Modify CG in three ways:

1. Look out for signs that $H \nsucc 0$.
2. Check whether we left the TR.
3. Seize opportunities to stop early.

$T_x M$




$v_{n-1} + t\, p_{n-1}$ with $t \geq 0$ s.t. $\|v_{n-1} + t\, p_{n-1}\|_x^2 = \Delta^2$

# Comments on RTR-tCG

- The first iterate of tCG is the Cauchy step, then it only gets better.
- Close to a strict minimizer, tCG makes essentially Newton steps.
- $H$ can be something else than $\mathrm{Hess} f(x)$, e.g., finite differences.
- Often need fewer iterations than GD b/c work harder at each $x_k$.
  $\rightarrow$ This means fewer retractions too + can reuse compute at $x_k$.
- It's useful to ensure $p_{n-1}$ is numerically in $\mathrm{T}_x\mathcal{M}$ periodically.