Chapter 13
Gradient Sampling Methods

One of the newest approaches in general NSO is to use gradient sampling algorithms developed by Burke et al. [51, 52]. The *gradient sampling method* (GS) is a method for minimizing an objective function that is locally Lipschitz continuous and smooth on an open dense subset $D \subset \mathbb{R}^n$. The objective may be nonsmooth and/or nonconvex. The GS may be considered as a stabilized steepest descent algorithm. The central idea behind these techniques is to approximate the subdifferential of the objective function through random sampling of gradients near the current iteration point. The ongoing progress in the development of gradient sampling algorithms (see e.g. [67]) suggests that they may have potential to rival bundle methods in the terms of theoretical might and practical performance. However, here we introduce only the original GS [51, 52].

13.1 Gradient Sampling Method

Let f be a locally Lipschitz continuous function on $\mathbb{R}^n$, and suppose that f is smooth on an open dense subset $D \subset \mathbb{R}^n$. In addition, assume that there exists a point $\bar{\boldsymbol{x}}$ such that the level set $\text{lev}_{f(\bar{\boldsymbol{x}})} = \{\boldsymbol{x} \mid f(\boldsymbol{x}) \leq f(\bar{\boldsymbol{x}})\}$ is compact.

At a given iterate $\boldsymbol{x}_k$ the gradient of the objective function is computed on a set of randomly generated nearby points $\boldsymbol{u}_{kj}$ with $j \in \{1, 2, \ldots, m\}$ and $m > n + 1$. This information is utilized to construct a search direction as a vector in the convex hull of these gradients with the shortest norm. A standard line search is then used to obtain a point with lower objective function value. The stabilization of the method is controlled by the *sampling radius* ε_k used to sample the gradients.

The pseudo-code of the GS is the following:

A. Bagirov et al., *Introduction to Nonsmooth Optimization*,
DOI: 10.1007/978-3-319-08114-4_13

PROGRAM GS
INITIALIZE $\boldsymbol{x}_0 \in \text{lev}_{f(\bar{\boldsymbol{x}})} \cap D$, $\varepsilon_0 > 0$, $m > n+1$, $\nu_0 \geq 0$, $\theta, \mu \in (0,1]$
and $\alpha, \beta \in (0,1)$;
Set $k = 0$;
WHILE the termination condition is not met
GRADIENT SAMPLING
Sample $\boldsymbol{u}_{k1}, \dots \boldsymbol{u}_{km}$ from $\bar{B}(\boldsymbol{x}; 1)$;
Set $\boldsymbol{x}_{k0} = \boldsymbol{x}_k$ and $\boldsymbol{x}_{kj} = \boldsymbol{x}_k + \epsilon_k \boldsymbol{u}_{kj}$ for $j = 1, \dots, m$;
IF $\boldsymbol{x}_{kj} \notin D$ for some j STOP;
Set $G_k = \{\nabla f(\boldsymbol{x}_{k1}), \nabla f(\boldsymbol{x}_{k2}), \dots, \nabla f(\boldsymbol{x}_{km})\}$;
END GRADIENT SAMPLING
Compute $\boldsymbol{g}_k = \text{argmin}_{\boldsymbol{g} \in G_k} \|\boldsymbol{g}\|^2$;
IF $\nu_k = \|\boldsymbol{g}_k\| = 0$ STOP with the final solution $\boldsymbol{x}_k$;
IF $\|\boldsymbol{g}_k\| > \nu_k$ THEN
Set $\nu_{k+1} = \nu_k$ and $\varepsilon_{k+1} = \varepsilon_k$;
Compute the search direction $\boldsymbol{d}_k = -\boldsymbol{g}_k / \|\boldsymbol{g}_k\|$;
Find the step size $t_k = \max \alpha^p$ such that
$f(\boldsymbol{x}_k + \alpha^p \boldsymbol{d}_k) < f(\boldsymbol{x}_k) - \beta \alpha^p \|\boldsymbol{g}_k\|$ and $p \in \{1, 2, \dots\}$;
ELSE
Set $t_k = 0$, $\nu_{k+1} = \theta \nu_k$, and $\varepsilon_{k+1} = \mu \varepsilon_k$;
END IF
IF $\boldsymbol{x}_k + t_k \boldsymbol{d}_k \in D$ THEN Set $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + t_k \boldsymbol{d}_k$;
ELSE
Let $\hat{\boldsymbol{x}}^k$ be any point on $\bar{B}(\boldsymbol{x}; \varepsilon_k)$ satisfying $\hat{\boldsymbol{x}}_k + t_k \boldsymbol{d}_k \in D$
and $f(\hat{\boldsymbol{x}}_k + t_k \boldsymbol{d}_k) < f(\hat{\boldsymbol{x}}_k) - \beta t_k \|\boldsymbol{g}_k\|$ (such a point exists
due to continuity of f);
Set $\boldsymbol{x}_{k+1} = \hat{\boldsymbol{x}}_k + t_k \boldsymbol{d}_k$;
END IF
Set $k = k + 1$;
END WHILE
RETURN final solution $\boldsymbol{x}_k$;
END PROGRAM GS

Note that the probability to obtain a point $\boldsymbol{x}_{kj} \notin D$ is zero in the above algorithm. In addition, it is reported in [52] that it is highly unlikely to have $\boldsymbol{x}_k + t_k \boldsymbol{d}_k \notin D$.

The GS algorithm may be applied to any function $f : \mathbb{R}^n \to \mathbb{R}$ that is continuous on $\mathbb{R}^n$ and differentiable almost everywhere. Furthermore, it has been shown that when f is locally Lipschitz continuous, smooth on an open dense subset D of $\mathbb{R}^n$, and has bounded level sets, the cluster point $\bar{\boldsymbol{x}}$ of the sequence generated by the GS with fixed ε is ε-stationary with probability 1 (that is, $\mathbf{0} \in \partial_\varepsilon^G f(\bar{\boldsymbol{x}})$, see also Definition 3.3 in Part I). In addition, if f has a unique ε-stationary point $\bar{\boldsymbol{x}}$, then the set of all cluster points generated by the algorithm converges to $\bar{\boldsymbol{x}}$ as ε is reduced to zero.