## 1.1 Subspace Methods

The basic idea of subspace methods is generating a sequence of subspaces $\mathcal{V}_m$ with increasing dimension $m$ and finding a vector $\boldsymbol{x}_m$ within each of these subspaces that is in some sense an *optimal* approximation in $\mathcal{V}_m$ to the solution $\boldsymbol{x}^*$ of the entire problem. Clearly, this optimality measure has to guarantee that we choose $\boldsymbol{x}_m = \boldsymbol{x}^*$ if $\boldsymbol{x}^* \in \mathcal{V}_m$ (at the latest if $m = n$). Therefore, designing a subspace method is subdivided in two tasks. First we have to define the sequence of subspaces and secondly we have to decide in which way we select a vector out of each subspace, i.e., which condition we provide to $\boldsymbol{x}_m \in \mathcal{V}_m$ in order to get a *good* approximation for $\boldsymbol{x}^*$ [20].

Let us assume for the moment that we already have chosen these subspaces $\mathcal{V}_m$ and now look for a criterion to select $\boldsymbol{x}_m \in \mathcal{V}_m$. From approximation theory we know that an optimal subspace approximation $\boldsymbol{x}_m$ is characterized by the fact that the error $\boldsymbol{x}^* - \boldsymbol{x}_m$ stays orthogonal on the subspace where $\boldsymbol{x}_m$ is chosen from, i.e.,

$$\boldsymbol{x}^* - \boldsymbol{x}_m \perp \mathcal{V}_m \tag{1.1}$$

must hold. Unfortunately, we do not know $\boldsymbol{x}^* - \boldsymbol{x}_m$ since we do not know the exact solution $\boldsymbol{x}^*$. However, we can compute the residual $\boldsymbol{r}_m := \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m$ which in some sense also is a measure for the *quality* of $\boldsymbol{x}_m$. Thus, a first idea might be simply to replace $\boldsymbol{x}^* - \boldsymbol{x}_m$ by $\boldsymbol{r}_m$ in (1.1).

Since we have

$$\boldsymbol{x}^* - \boldsymbol{x}_m = \boldsymbol{A}^{-1}\boldsymbol{b} - \boldsymbol{A}^{-1}\boldsymbol{A}\boldsymbol{x}_m = \boldsymbol{A}^{-1}\boldsymbol{r}_m$$

it might be advantageous to choose $\boldsymbol{x}_m \in \mathcal{V}_m$ satisfying

$$\boldsymbol{A}^{-1}\boldsymbol{r}_m \perp \mathcal{V}_m \quad \Leftrightarrow \quad \boldsymbol{r}_m \perp \boldsymbol{A}\mathcal{V}_m, \tag{1.2}$$

that is, $\boldsymbol{r}_m$ may not stay orthogonal on $\mathcal{V}_m$ but on another subspace, say $\mathcal{W}_m$.

We fix these ideas in the following definition.

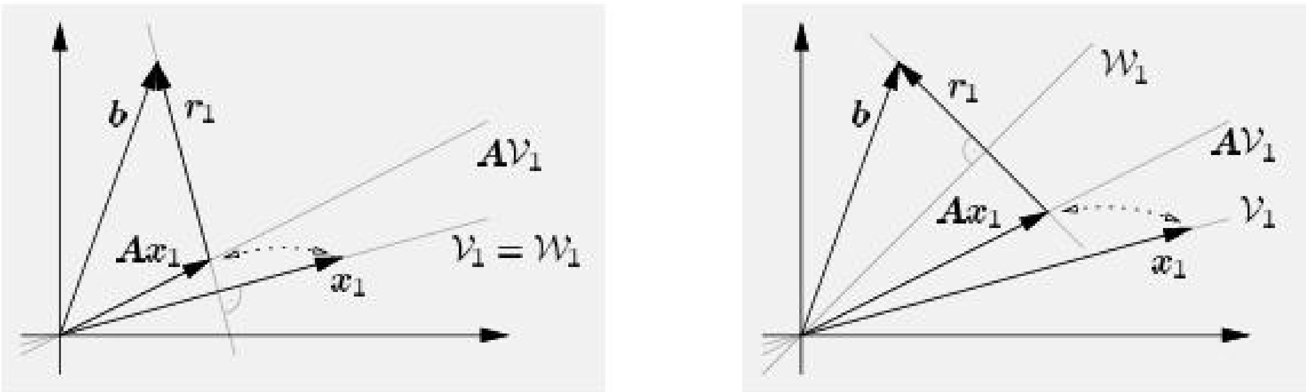
**Definition 1.1** *A projecting method for solving a linear system $\boldsymbol{Ax} = \boldsymbol{b}$ is a procedure that constructs approximate solutions $\boldsymbol{x}_m \in \mathcal{V}_m$ under the constraint*

$$\boldsymbol{r}_m = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m \perp \mathcal{W}_m \tag{1.3}$$

*where $\mathcal{V}_m$ and $\mathcal{W}_m$ are $m$-dimensional subspaces of $\mathbb{R}^n$.*

*In the case $\mathcal{W}_m = \mathcal{V}_m$ we have an orthogonal projecting method and (1.3) is called a Galerkin condition whereas the general case ($\mathcal{W}_m \neq \mathcal{V}_m$) is called a skew or oblique projecting method with a Petrov-Galerkin condition in equation (1.3).*

In Figure 1.1 we illustrate the case $n = 2$, $m = 1$. Given $\mathcal{V}_1$ we select $\boldsymbol{x}_1 \in \mathcal{V}_1$ to satisfy $\boldsymbol{r}_1 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_1 \perp \mathcal{V}_1$ or $\boldsymbol{r}_1 \perp \mathcal{W}_1$ in the case of skew projections.




(a) orthogonal projection

(b) skew projection

**Figure 1.1:** Projecting methods. Given $\mathcal{V}_1$ we compute $\boldsymbol{A}\mathcal{V}_1 = \{\boldsymbol{Ax} \mid \boldsymbol{x} \in \mathcal{V}_1\}$ and then select $\boldsymbol{x}_1 \in \mathcal{V}_1$ such that $\boldsymbol{r}_1 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_1 \perp \mathcal{V}_1$, respectively $\boldsymbol{r}_1 \perp \mathcal{W}_1$ in the case of skew projections.

Another way to characterize an optimal approximation is by its error norm. That means $\boldsymbol{x}_m \in \mathcal{V}_m$ is called optimal if $\|\boldsymbol{x}^* - \boldsymbol{x}_m\|$ minimizes $\|\boldsymbol{x}^* - \boldsymbol{x}\|$ for all $\boldsymbol{x} \in \mathcal{V}_m$. Again we have to replace $\boldsymbol{x}^* - \boldsymbol{x}_m$ by $\boldsymbol{r}_m$ (because we generally do not know $\boldsymbol{x}^*$) and fix our ideas in the following definition.

**Definition 1.2** *A norm minimizing method for solving a linear system $\boldsymbol{Ax} = \boldsymbol{b}$ is a procedure that constructs approximate solutions $\boldsymbol{x}_m \in \mathcal{V}_m$ under the constraint*

$$\|\boldsymbol{r}_m\|_2 = \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_m\|_2 = \min_{\boldsymbol{x} \in \mathcal{V}_m} \{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2\} \tag{1.4}$$

*where $\mathcal{V}_m$ is an $m$-dimensional subspace of $\mathbb{R}^n$.*

## 1.2 Generating Krylov Spaces

In this section we focus on the question of how to select the subspaces $\mathcal{V}_m$. To take a possibly given initial guess $\boldsymbol{x}_0$ into account, we allow $\mathcal{V}_m$ to be an affine subspace,



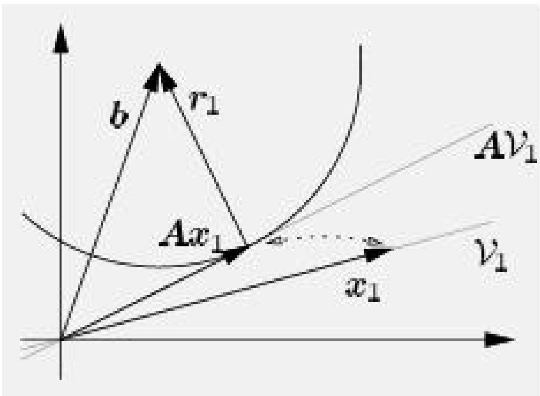



**Figure 1.2:** Norm minimizing methods. Given $\mathcal{V}_1$ we compute $\boldsymbol{A}\mathcal{V}_1 = \{\boldsymbol{A}\boldsymbol{x} \mid \boldsymbol{x} \in \mathcal{V}_1\}$ and then select $\boldsymbol{x}_1 \in \mathcal{V}_1$ to satisfy $\|\boldsymbol{r}_1\|_2 = \|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_1\|_2 = \min_{\boldsymbol{x}\in\mathcal{V}_1}\{\|\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}\|_2\}$.

i.e., $\mathcal{V}_m = \boldsymbol{x}_0 + \tilde{\mathcal{V}}_m$. There are various possibilities to choose $\tilde{\mathcal{V}}_m$ but it turns out that using Krylov subspaces has several advantageous properties as we will see later on.

**Definition 1.3** *A Krylov subspace method is a projecting or a norm minimizing method (see Definitions 1.1 and 1.2) to solve a linear system* $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ *where the subspaces* $\tilde{\mathcal{V}}_m$ *are chosen as Krylov subspaces*

$$\tilde{\mathcal{V}}_m = \mathcal{K}_m(\boldsymbol{A}, \boldsymbol{r}_0) := \operatorname{span}\{\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^{m-1}\boldsymbol{r}_0\}, \quad m = 1, 2, \ldots \tag{1.5}$$

*with* $\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_0$.

Since we intend to work with vectors out of $\mathcal{K}_m(\boldsymbol{A}, \boldsymbol{r}_0)$ we have to find a handy representation for them. One of the most powerful tricks in linear algebra that often simplifies problems significantly, is to find a *suitable* basis of the vector space we have to work with. Often orthonormal bases are a good choice but it is generally an enormous amount of work to compute one. The Arnoldi algorithm (Section 1.2.1) and the Lanczos algorithm (Section 1.2.2) are methods to compute orthonormal bases of Krylov spaces. In Section 1.2.3 we will see that there is a cheaper way to get a useful basis, although it is not orthonormal anymore.