Fig. 10.2  Forward-backward versus Beck-Teboulle: As in Example 10.12, let $C$ and $D$ be two closed convex sets and consider the problem (10.30) of finding a point $x_\infty$ in $C$ at minimum distance from $D$. Let us set $f_1 = \iota_C$ and $f_2 = d_{D/2}^2$. Top: The forward–backward algorithm with $\gamma_n \equiv 1$ and $\lambda_n \equiv 1$. As seen in Example 10.12, it reduces to the alternating projection method (10.31). Bottom: The Beck-Teboulle algorithm.
Fig. 10.3 Forward-backward versus Douglas–Rachford: As in Example 10.12, let $C$ and $D$ be two closed convex sets and consider the problem (10.30) of finding a point $x_\infty$ in $C$ at minimum distance from $D$. Let us set $f_1 = \iota_C$ and $f_2 = d_2^D/2$. Top: The forward–backward algorithm with $\gamma_n \equiv 1$ and $\lambda_n \equiv 1$. As seen in Example 10.12, it assumes the form of the alternating projection method (10.31).

Bottom: The Douglas–Rachford algorithm with $\gamma = 1$ and $\lambda_m \equiv 1$. Table 10.1.xii yields prox$_{f_1} = \Pi_C$ and Table 10.1.vi yields prox$_{f_2} : x \mapsto (x + \Pi_D x)/2$. Therefore the updating rule in Algorithm 10.15 reduces to $x_n = (y_n + \Pi_D y_n)/2$ and $y_{n+1} = \Pi_C(2x_n - y_n) + y_n - x_n = \Pi_C(\Pi_D y_n) + y_n - x_n$. 

$x_0$ $x_1$ $x_2$ $x_3$ ... $x_\infty$ $y_0$ $y_1$ $y_2$ $y_3$ $y_4$...
such techniques were introduced in \cite{110, 111} and have been used in the context of convex feasibility problems in \cite{10, 43, 45}. To this end, observe that \eqref{10.53} can be rewritten in $H$ as
\begin{equation}
\minimize_{x_1, \ldots, x_m \in H \atop x_1 = \cdots = x_m} f_1(x_1) + \cdots + f_m(x_m).
\tag{10.55}
\end{equation}
If we denote by $x = (x_1, \ldots, x_m)$ a generic element in $H$, \eqref{10.55} is equivalent to
\begin{equation}
\minimize_{x \in H} \iota_D(x) + f(x),
\tag{10.56}
\end{equation}
where
\begin{align}
D &= \{(x, \ldots, x) \in H \mid x \in \mathbb{R}^N\} \\
f : x &\mapsto f_1(x_1) + \cdots + f_m(x_m).
\tag{10.57}
\end{align}
We are thus back to a problem involving two functions in the larger space $H$. In some cases, this observation makes it possible to obtain convergent methods from the algorithms discussed in the preceding sections. For instance, the following parallel algorithm was derived from the Douglas–Rachford algorithm in \cite{54} (see also \cite{49} for further analysis and connections with Spingarn’s splitting method \cite{120}).

**Algorithm 10.27 (Parallel proximal algorithm (PPXA))**
Fix $\varepsilon \in ]0, 1[\setminus \gamma > 0$, $(\omega_i)_{1 \leq i \leq m} \in ]0, 1]^m$ such that
\begin{equation}
\sum_{i=1}^m \omega_i = 1, \quad y_{i,0} \in \mathbb{R}^N, \ldots, y_{m,0} \in \mathbb{R}^N
\end{equation}
Set $x_0 = \sum_{i=1}^m \omega_i y_{i,0}$
For $n = 0, 1, \ldots$
\begin{equation}
\begin{aligned}
&f_n = \sum_{i=1}^m \omega_i p_{i,n} \\
&\varepsilon \leq \lambda_n \leq 2 - \varepsilon \\
&\text{For } i = 1, \ldots, m
\begin{cases}
\lambda_{i,n} = \lambda_{i,n}\left(2p_n - x_n - p_{i,n}\right) \\
x_{i,n+1} = x_{i,n} + \lambda_{i,n} (p_n - x_n).
\end{cases}
\end{aligned}
\end{equation}

**Proposition 10.28** \cite{54} Every sequence $(x_n)_{n \in \mathbb{N}}$ generated by Algorithm 10.27 converges to a solution to Problem 10.26.

**Example 10.29 (image recovery)** In many imaging problems, we record an observation $y \in \mathbb{R}^M$ of an image $z \in \mathbb{R}^K$ degraded by a matrix $L \in \mathbb{R}^{M \times K}$ and corrupted by noise. In the spirit of a number of recent investigations (see \cite{37} and the references therein), a tight frame representation of the images under consideration can be used. This representation is defined through a synthesis matrix $F^\top \in \mathbb{R}^{K \times N}$ (with $K \leq N$) such that $F^\top F = \nu I$, for some $\nu \in ]0, +\infty[$. Thus, the original image can be written
\begin{equation}
(z_1, \ldots, z_m) = F^\top (f_1, \ldots, f_m).
\end{equation}