1 Linear Programming Preconditioners for the Nonlinear Interval Gauss-Seidel Method with Midpoint Predictor

In applying interval techniques to nonlinear systems, we obtain and bound solutions of transformed interval linear systems of the form

$$\boldsymbol{A}(\boldsymbol{X} - X) = \boldsymbol{B}, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{IR}m \times n$, $X \in \boldsymbol{X}$, and $\boldsymbol{B}$ is usually a vector of narrow intervals or scalars.

In handling such systems, it is often useful, and sometimes necessary, to precondition the system by multiplication with a matrix Y. For the case where $m = n$, some common choices for Y appearing in the literature are the inverse of the midpoint matrix of $\boldsymbol{A}$ and the inverse of the Jacobian matrix for the function at X.

In [] Kearfott presented a method using linear programming for computing preconditioning matrices in a row-wise fashion for the system 1 when $X = m(\boldsymbol{X})$ and $w(\boldsymbol{B}) = 0$ (or negligible). These preconditioners were optimal in a certain sense when using the interval Gauss-Seidel procedure.

In this chapter, we expand on that work by presenting more efficient formulations of the related linear programming problems, and by providing additional theoretical results. We will also mention some empirical observations made concerning these linear programming preconditioners.

1.1 A Row-wise View of Preconditioners

Consider the Gauss-Seidel step in the k-th coordinate for the system 1,

$$\tilde{\boldsymbol{x}}_k = x_k - \left(\sum_{i=1}^{m} y_{k,i}\boldsymbol{b}_i + \sum_{\substack{j=1 \\ j\neq k}}^{n} (\boldsymbol{x}_j - x_j) \sum_{i=1}^{m} y_{k,i}\boldsymbol{a}_{i,j} \right) \Bigg/ \left(\sum_{i=1}^{m} y_{k,i}\boldsymbol{a}_{i,k} \right). \tag{2}$$

Viewed as an isolated step, the computation of $\tilde{\boldsymbol{x}}_k$ depends on no other elements of Y but those in the k-th row. Hence, when dealing with the interval Gauss-Seidel method, it is reasonable to consider preconditioning rows individually instead of considering the entire preconditioning matrix Y.

In fact, we are really dealing with equivalence classes of preconditioning rows since

Lemma 1.1 (Kearfott) *When applying the Gauss-Seidel step in the k-th coordinate, every nonzero multiple of the k-th row of Y yields the same $\tilde{\boldsymbol{x}}_k$ (in exact arithmetic).*

Hereafter, we will denote a preconditioning row for the Gauss-Seidel step in the k-th coordinate by Y_k, and we define equivalence of preconditioning rows as follows.

Definition 1.1 *Two preconditioning rows for the Gauss-Seidel step in the k-th coordinate are equivalent iff they are nonzero scalar multiples of each other.*

We also will refer to the numerator and the denominator of the fraction on the right hand side of equation 2 by $\boldsymbol{n}_k(Y_k) = [\underline{n}_k, \overline{n}_k]$ and $\boldsymbol{d}_k(Y_k) = [\underline{d}_k, \overline{d}_k]$ respectively.

1.2 Classification of Preconditioning Rows

Two general disjoint classes of useful preconditioning rows are now defined.

Definition 1.2 *A preconditioning row Y_k is a C-preconditioner iff $0 \notin \boldsymbol{d}_k(Y_k)$. Furthermore, Y_k is a normal C-preconditioner iff $\underline{d}_k = 1$.*

Definition 1.3 *A preconditioning row Y_k is an E-preconditioner iff $0 \in \boldsymbol{d}_k(Y_k)$ and $0 \notin \boldsymbol{n}_k(Y_k)$. Furthermore, Y_k is a normal E-preconditioner iff $\underline{n}_k = 1$.*

Note that all useful preconditioning rows are either C-preconditioners or E-preconditioners. A preconditioner not falling into either of these catagories would necessarily have $0 \in \boldsymbol{d}_k(Y_k)$ and $0 \in \boldsymbol{n}_k(Y_k)$, resulting in $\tilde{\boldsymbol{x}}_k$ being set to $(-\infty, \infty)$.

In general, determining the existence of an E-preconditioner is a nontrivial task. However, we have the following simple test for the existence of a C-preconditioner.

Lemma 1.2 (Hu) *There exists a C-preconditioner Y_k iff at least one element of the k-th column of $\boldsymbol{A}$ does not contain 0.*

We also define the following specific C-preconditioners.

Definition 1.4 *The C-preconditioner Y_k is <u>width-optimal</u>, or a <u>C^W-preconditioner</u>, iff it minimizes the width of $\tilde{\boldsymbol{x}}_k$ over all C-preconditioners.*

Definition 1.5 *The C-preconditioner Y_k is <u>left-optimal</u>, or a <u>C^L-preconditioner</u>, iff it maximizes the left endpoint of $\tilde{\boldsymbol{x}}_k$ over all C-preconditioners.*

Definition 1.6 *The C-preconditioner Y_k is <u>right-optimal</u>, or a <u>C^R-preconditioner</u>, iff it minimizes the right endpoint of $\tilde{\boldsymbol{x}}_k$ over all C-preconditioners.*

Analogous specific E-preconditioners will be defined later.

1.3 Some Important Identities

Lemma 1.3 *Let $r \in \Re$ and $\boldsymbol{x}, \boldsymbol{r} \in \Re$, and denote the positive and negative parts of r by $r = \max\{r, 0\}$ and $r = \max\{-r, 0\}$ respectively. Then the following are true:*

1. $r\boldsymbol{x} = [r - r, r - r]$.
2. $w[\boldsymbol{r}(\boldsymbol{x} - m(\boldsymbol{x}))] = |\boldsymbol{r}|w(\boldsymbol{x})$.
3. $|\boldsymbol{x}| = \max\{-, \}$.
4. $\boldsymbol{r}(\boldsymbol{x} -) = [-\underline{r}w(\boldsymbol{x}), \underline{r}w(\boldsymbol{x})]$.
5. $\boldsymbol{r}(\boldsymbol{x} -) = [-\underline{r}w(\boldsymbol{x}), \underline{r}w(\boldsymbol{x})]$.
6. $|\boldsymbol{x}| = \delta[- + (+)] + (1 - \delta)[+(+)]$ *for any* $\delta \in [0, 1]$.

Lemma 1.4 *Let Y_k be a C-preconditioner and suppose that for all $1 \le i \le n$ we have $x_i \in \{_i, m(\boldsymbol{x}_i), _i\}$. Define the following:*

$$
\begin{aligned}
\mathcal{V}_M &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = m(\boldsymbol{x}_j) \text{ and } w(\boldsymbol{x}_j) \ne 0\}. \\
\mathcal{V}_L &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}. \\
\mathcal{V}_R &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}.
\end{aligned}
$$

Then

$$
\begin{aligned}
\boldsymbol{n}_k(Y_k) \;=\; & \sum_{i=1}^{m} y_{k,i} m(\boldsymbol{b}_i) + \sum_{i=1}^{m} |y_{k,i}| w(\boldsymbol{b}_i) \left[-\tfrac{1}{2}, \tfrac{1}{2}\right] \\
& + \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) \left|\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}\right| \left[-\tfrac{1}{2}, \tfrac{1}{2}\right] \\
& + \sum_{j \in \mathcal{V}_L} w(\boldsymbol{x}_j) \left[-\left(\underline{\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}}\right), \left(\overline{\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}}\right)\right] \\
& + \sum_{j \in \mathcal{V}_R} w(\boldsymbol{x}_j) \left[-\left(\overline{\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}}\right), \left(\underline{\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}}\right)\right].
\end{aligned}
$$

If $x_i = m(\boldsymbol{x}_i)$ for all $1 \le i \le n$, then

$$w(\boldsymbol{n}_k(Y_k)) = \sum_{i=1}^{m} |y_{k,i}| w(\boldsymbol{b}_i) + \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) |\textstyle\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j}|.$$

Lemma 1.5 *For all preconditioners Y_k and all $1 \le j \le n$ we have*

$$\sum_{i=1}^{m} y_{k,i} \boldsymbol{a}_{i,j} = \left[\sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}}\right), \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}}\right)\right].$$

For any preconditioner Y_k we have

$$\underline{d}_k = \sum_{i=1}^{m} \left(y_{k,i_{i,k}} - y_{k,i_{i,k}}\right).$$

1.4 Computation of Width-Optimal C-Preconditioners Using Linear Programming

In this section, assume that a C-preconditioner exists for the interval Gauss-Seidel step in the k-th coordinate.

By Lemma 1.1, finding a (normal) preconditioner Y_k involves solving the nonlinear optimization problem

$$\min_{\underline{d}_k = 1} w(\boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)). \tag{3}$$

However, Kearfott [] observed the following Given system 1, suppose that Y_k is a normal C-preconditioner and $\tilde{\boldsymbol{x}}_k$ is computed by equation 2. Then the following hold (in exact arithmetic):

1. If $0 \in \boldsymbol{n}_k(Y_k)$, then $w(\tilde{\boldsymbol{x}}_k) = w(\boldsymbol{n}_k(Y_k))$.

2. If $\overline{n}_k < 0$, then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < \min\{-\underline{n}_k, ({}_k - x_k)\}$.

3. If $0 < \underline{n}_k$, then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < \min\{\overline{n}_k, (x_k - {}_k)\}$.

In particular, if $0 \notin \boldsymbol{n}_k(Y_k)$ and $x_k = m(\boldsymbol{x}_k)$ then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < w(\boldsymbol{x}_k)/2$.

Hence, a normal C-preconditioner Y_k which solves

$$\min_{\underline{d}_k = 1} w(\boldsymbol{n}_k(Y_k)), \tag{4}$$

is a normal preconditioner if $0 \in \boldsymbol{n}_k(Y_k)$. Otherwise, when using the midpoint predictor at least half of $\boldsymbol{x}_k$ will be eliminated after intersection with $\tilde{\boldsymbol{x}}_k$. In addition, as we shall show, problem 4 can be stated as a linear programming problem. These facts make solving problem 4 an attractive alternative to solving problem 3 when attempting to find preconditioners.

In [], Kearfott used the first three identities of Lemma 1.3 and the representation of $w(\boldsymbol{n}_k(Y_k))$ from Corollary 1.3 to construct a linear programming problem related to problem 4 for the case $\boldsymbol{A} \in \Re^{n \times n}$, $X = m(\boldsymbol{X})$, and $w(\boldsymbol{B}) = 0$ (or negligible). In standard form, this problem contained $5n - 3$ variables and $2n-1$ constraints. If we define $\mathcal{U} = \{j \mid 1 \le j \le n \text{ and } j \ne k\}$, we may state Kearfott's formulation as

$$\begin{array}{lrcll} \text{minimize} & \displaystyle\sum_{j \in \mathcal{U}} v'_j w(\boldsymbol{x}_j) & & & \\ \text{subject to} & \displaystyle v''_j - v'_j - \sum_{i=1}^{n} (y'_i{}_{i,j} - y''_i{}_{i,j}) & = & 0, & j \in \mathcal{U}, \\ & \displaystyle v'''_j - v'_j + \sum_{i=1}^{n} (y'_i{}_{i,j} - y''_i{}_{i,j}) & = & 0, & j \in \mathcal{U}, \\ & \displaystyle \sum_{i=1}^{n} (y'_i{}_{i,k} - y''_i{}_{i,k}) & = & 1, & \\ \text{and} & v'_j \ge 0, \quad v''_j \ge 0, \quad v'''_j \ge 0, & & & j \in \mathcal{U}, \\ & y'_i \ge 0, \quad y''_i \ge 0, & & & 1 \le i \le n. \end{array} \tag{5}$$

After solving problem 5 for $(y'; y''; v'; v''; v''')$, a preconditioner Y_k is computed by setting $y_{k,i} y'_i - y''_i$, for $1 \leq i \leq n$.

It was shown that, under certain conditions, the preconditioner Y_k obtained by solving problem 5 also solved problem 4. Kearfott also conjectured, based on empirical evidence, that those conditions were always satisfied when using the simplex method with exact arithmetic. Finally, it was shown that the linear programming problem was feasible iff a C-preconditioner existed.

1.5 An Improved Linear Programming Formulation for Computing Width-Optimal C-Preconditioners

Lemma 1.6 *Let $\mathcal{V} = \{j \mid 1 \leq j \leq n,\ \ j \neq k,\ \text{and } w(\boldsymbol{x}_j) \neq 0\}$ and choose $\delta_j \in [0,1]$ for all $j \in \mathcal{V}$. If $X = m(\boldsymbol{X})$ then*

$$\begin{aligned} w(\boldsymbol{n}_k(Y_k)) = \ & \sum_{i=1}^{m} (y_{k,i} + y_{k,i}) w(\boldsymbol{b}_i) + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j) \delta_j \left[v_j - \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\ & + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j)(1 - \delta_j) \left[v_j + \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \end{aligned}$$

where

$$v_j = \sum_{i=1}^{m} (y_{k,i} - y_{k,i})({}_{i,j} + {}_{i,j}), \quad j \in \mathcal{V}.$$

Based on this result, we construct a linear programming problem related to problem 4 with $X = m(\boldsymbol{X})$, which has several advantages over problem 5. For $\boldsymbol{A} \in \mathbb{IR}m \times n$, this problem contains at most $2(n + m - 1)$ variables and n constraints when in standard form. By defining $\mathcal{V} = \{j \mid 1 \leq j \leq n,\ \ j \neq k,\ \text{and } w(\boldsymbol{x}_j) \neq 0\}$ and choosing $\delta_j \in [0,1]$ for all $j \in \mathcal{V}$, we may state this linear programming problem as

$$
\begin{array}{lrcl}
\text{minimize} & \multicolumn{3}{l}{W_k(y';y'';v';v'') =} \\
& \multicolumn{3}{l}{\displaystyle\sum_{i=1}^{m}(y'_i + y''_i)w(\boldsymbol{b}_i) + \sum_{j\in\mathcal{V}} w(\boldsymbol{x}_j)\delta_j \left[v'_j - \sum_{i=1}^{m} (y'_i{}_{i,j} - y''_i{}_{i,j}) \right]} \\
& \multicolumn{3}{l}{\displaystyle + \sum_{j\in\mathcal{V}} w(\boldsymbol{x}_j)(1-\delta_j) \left[v''_j + \sum_{i=1}^{m} (y'_i{}_{i,j} - y''_i{}_{i,j}) \right]} \\
\text{subject to} & \displaystyle v''_j - v'_j + \sum_{i=1}^{m}(y'_i - y''_i)({}_{i,j}+{}_{i,j}) & = & 0, \quad j \in \mathcal{V}, \\
& \displaystyle \sum_{i=1}^{m} (y'_i{}_{i,k} - y''_i{}_{i,k}) & = & 1, \\
\text{and} & \multicolumn{3}{c}{v'_j \ge 0, \quad v''_j \ge 0, \quad j \in \mathcal{V},} \\
& \multicolumn{3}{c}{y'_i \ge 0, \quad y''_i \ge 0, \quad 1 \le i \le n.}
\end{array}
\tag{6}
$$

After solving problem 6 for $(y';y'';v';v'')$, a preconditioner Y_k is obtained as follows:

Definition 1.7 *If $(y';y'';v';v'')$ is a feasible point of problem 6, we define the preconditioner generated by $(y';y'';v';v'')$ to be $Y_k = H(y';y'';v';v'')$, which is computed by setting $y_{k,i} y'_i - y''_i$ for $1 \le i \le n$.*

In the analysis of problem 6, the following type of feasible point is of particular interest.

Definition 1.8 *A feasible point $(y';y'';v';v'')$ of problem 6 is <u>normal</u> iff $y'_i y''_i = 0$ for all $1 \le i \le n$ and $v'_j v''_j = 0$ for all $j \in \mathcal{V}$.*

We now show that preconditioners generated by feasible points of problem 6 must be C-preconditioners.

Lemma 1.7 *Let $(y';y'';v';v'')$ be a feasible point of problem 6, and let Y_k be its generated preconditioner. Then Y_k is a C-preconditioner with $\underline{d}_k \ge 1$. Furthermore, if $(y';y'';v';v'')$ is a normal feasible point, then Y_k is a normal C-preconditioner ($\underline{d}_k = 1$).*

Define $\alpha_i = \min\{y'_i, y''_i\} \geq 0$ for $1 \leq i \leq n$. Then for $1 \leq i \leq n$ we have $y_{k,i} = y'_i - \alpha_i$ and $y_{k,i} = y''_i - \alpha_i$. Hence, by Lemma 1.5,

$$\begin{aligned}
\underline{d}_k &= \sum_{i=1}^{m} ((y'_i - \alpha_i)_{i,k} - (y''_i - \alpha_i)_{i,k}) \\
&= \sum_{i=1}^{m} (y'_i{}_{i,k} - y''_i{}_{i,k}) + \sum_{i=1}^{m} \alpha_i \,({}_{i,k} - {}_{i,k}) \\
&= 1 + \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,k}) \\
&\geq 1.
\end{aligned}$$

In particular, if $(y'; y''; v'; v'')$ is a normal feasible point, then $\alpha_i = 0$ for all $1 \leq i \leq n$, and hence $\underline{d}_k = 1$.

Note that it is possible for two distinct feasible points to generate the same preconditioner. However, unlike arbitrary feasible points, normal feasible points have the following property.

Lemma 1.8 *No two distinct normal feasible points of problem 6 can generate equivalent preconditioners.*

The above results allow us to give the following characterization of the relationship between problem 4 and problem 6 for the case $X = m(\boldsymbol{X})$ when only *normal* feasible points of problem 6 are considered.

Lemma 1.9 *If $X = m(\boldsymbol{X})$ then H is a bijective function from the set of normal feasible points of problem 6 to the set of normal C-preconditioners such that for any normal feasible point $(y'; y''; v'; v'')$ and any normal C-preconditioner Y_k we have*

$$Y_k = H(y'; y''; v'; v'') \Rightarrow W_k(y'; y''; v'; v'') = w(\boldsymbol{n}_k(Y_k)).$$

Furthermore, $H(Y_k) = (Y_k; Y_k; v; v)$ where $v_j = \sum_{i=1}^{m} y_{k,i}({}_{i,j} + {}_{i,j}),\ \ j \in \mathcal{V}$.

By Lemma 1.6 and Corollary 1.3, every normal C-preconditioner Y_k generates a normal feasible point $H(Y_k) = (y'; y''; v'; v'')$ of problem 6 for which we have $W_k(y'; y''; v'; v'') = w(\boldsymbol{n}_k(Y_k))$. The relation $H(H(Y_k)) = Y_k$ follows easily from the definitions of H and H.

Conversely, by Lemma 1.7 every normal feasible point $(y'; y''; v'; v'')$ generates a normal C-preconditioner $H(y'; y''; v'; v'') = Y_k$. By Lemma 1.8 and the preceding case in this proof we must have that $H(H(y';y'';v';v'')) = (y';y'';v';v'')$.

It turns out that we need not concern ourselves with non-normal feasible points when dealing with problem 6. The following Theorem demonstrates this.

Lemma 1.10 *Let $(y';y'';v';v'')$ be a feasible point of problem 6. Then we may construct a normal feasible point $(z';z'';u';u'')$ associated with $(y';y'';v';v'')$ for which $W_k(z';z'';u';u'') \le W_k(y';y'';v';v'')$. Furthermore, $H(z';z'';u';u'')$ is a normal C-preconditioner equivalent to the C-preconditioner $H(y';y'';v';v'')$.*

Let Y_k be the C-preconditioner generated by $(y';y'';v';v'')$. For $j \in \mathcal{V}$ define $v_j = \sum_{i=1}^{m} y_{k,i}({}_{i,j} + {}_{i,j})$. Also, define $\alpha_i = \min\{y'_i, y''_i\} \ge 0$ for $1 \le i \le n$, and $\beta_j = \min\{v'_j, v''_j\} \ge 0$ for $j \in \mathcal{V}$. Then we have $y_{k,i} = y'_i - \alpha_i$ and $y_{k,i} = y''_i - \alpha_i$ for all $1 \le i \le n$, as well as $v_j = v'_j - \beta_j$ and $v_j = v''_j - \beta_j$ for all $j \in \mathcal{V}$. So, by Lemma 1.6 we have

$$
\begin{aligned}
w(\boldsymbol{n}_k(Y_k)) &= \sum_{i=1}^{m} (y_{k,i} + y_{k,i}) w(\boldsymbol{b}_i) + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j)\delta_j \left[v_j - \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
&\quad + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j)(1-\delta_j) \left[v_j + \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
&= \sum_{i=1}^{m} (y'_i + y''_i) w(\boldsymbol{b}_i) + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j)\delta_j \left[v'_j - \sum_{i=1}^{m} (y'_{i i,j} - y''_{i\,i,j}) \right] \\
&\quad + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j)(1-\delta_j) \left[v''_j + \sum_{i=1}^{m} (y'_{i i,j} - y''_{i\,i,j}) \right] \\
&\quad - \left\{ 2 \sum_{i=1}^{m} \alpha_i w(\boldsymbol{b}_i) + \sum_{j \in \mathcal{V}} w(\boldsymbol{x}_j) \left[\beta_j + \sum_{i=1}^{m} \alpha_i \left({}_{i,j} - {}_{i,j} \right) \right] \right\} \\
&\le W_k(y';y'';v';v'').
\end{aligned}
$$

Furthermore, we must have $0 \le w(\boldsymbol{n}_k(Y_k)) \le W_k(y';y'';v';v'')$ since $w(\boldsymbol{n}_k(Y_k))$ is the width of an interval.

By Lemma 1.7, $\underline{d}_k \ge 1$. Define $Z_k = \frac{1}{\underline{d}_k} Y_k$. Then Z_k is a normal C-preconditioner equivalent to Y_k. By Lemma 1.9, the point $(z';z'';u';u'') =$

$H(Z_k)$ is a normal feasible point for which

$$\begin{aligned} W_k(z';z'';u';u'') &= w(\boldsymbol{n}_k(Z_k)) \\ &= \frac{1}{\underline{d}_k} w(\boldsymbol{n}_k(Y_k)) \\ &\leq w(\boldsymbol{n}_k(Y_k)) \\ &\leq W_k(y';y'';v';v''). \end{aligned}$$

Finally, by Lemma 1.9, the normal C-preconditioner generated by $(z';z'';u';u'')$ is in fact Z_k.

Hence, for any feasible point of problem 6 there exists a normal feasible point having an equal or smaller objective function value and generating an equivalent (normal) C-preconditioner. This and the fact that we are interested in the generated preconditioners, and not the feasible points of problem 6 themselves, motivates the following definitions.

Definition 1.9 *Two feasible points of problem 6 are <u>equivalent</u> iff they generate equivalent preconditioners. An equivalence class containing a solution to problem 6 is called a <u>solution equivalence class</u>.*

By Lemma 1.8 and 1.10, we have the following result.

Lemma 1.11 *Each equivalence class of feasible points contains a unique normal feasible point. Furthermore, the minimum objective function value of problem 6 over a class is attained at the normal feasible point in that class.*

We are now able to prove the following theorem relating problem 4 and problem 6 for the case $X = m(\boldsymbol{X})$. If $X = m(\boldsymbol{X})$ then problem 4 and problem 6 are equivalent in the following sense:

1. Each of the problems is feasible iff a C-preconditioner exists.

2. There is a bijective function S from the set of normal C-preconditioners (feasible points of problem 4) to the set of equivalence classes of feasible points of problem 6 such that for any normal C-preconditioner Y_k and for any $(z';z'';u';u'') \in S(Y_k)$ we have

   $$w(\boldsymbol{n}_k(Y_k)) = W_k(y';y'';v';v'') \leq W_k(z';z'';u';u''),$$

   where $(y';y'';v';v'')$ is the unique normal feasible point in $S(Y_k)$.

3. The restriction $\tilde{S}$ of S to the set of solutions of problem 4 is a bijective function from that set to the set of solution equivalence classes of problem 6.

Recall that existence of a C-preconditioner implies existence of an equivalent normal C-preconditioner. Then clearly problem 4 is feasible iff a C-preconditioner exists, as it is a minimization problem over all normal C-preconditioners. Also, existence of a normal C-preconditioner implies existence of a (normal) feasible point for problem 6 by Lemma 1.6. Conversely, every feasible point of problem 6 generates a C-preconditioner by Lemma 1.7. Hence the first statement is proved.

Next, recall that the function H defined in Lemma 1.9 is a bijection between the set of normal C-preconditioners and the set of normal feasible points. Since each equivalence class of feasible points contains a unique normal feasible point by Lemma 1.11, we may define $S(Y_k)$ to be the equivalence class of feasible points containing $H(Y_k)$. Similarly, for any equivalence class E of feasible points of problem 6 we may define $S(E)$ to be the normal C-preconditioner $H(y';y'';v';v'')$, where $(y';y'';v';v'')$ is the unique normal feasible point contained in E. It follows that S must be bijective with inverse S. Then by Lemma 1.9, Lemma 1.10, and Lemma 1.11 we must have $w(\boldsymbol{n}_k(Y_k)) = W_k(y';y'';v';v'') \leq W_k(z';z'';u';u'')$, where Y_k is a normal C-preconditioner, $(y';y'';v';v'')$ is the unique normal feasible point in $S(Y_k)$, and $(z';z'';u';u'') \in S(Y_k)$ is arbitrary. This proves the second statement.

Now, let Y_k be a normal C-preconditioner which solves problem 4 and suppose that $S(Y_k)$ is not a solution equivalence class of problem 6. By Lemma 1.9, $H(Y_k)$ is a normal feasible point of problem 6 for which we have $w(\boldsymbol{n}_k(Y_k)) = W_k(H(Y_k))$. Since $S(Y_k)$ was not a solution equivalence class, there must exist a feasible point $(z';z'';u';u'')$ of problem 6 for which we have $W_k(z';z'';u';u'') < W_k(H(Y_k))$. By Lemma 1.10 we may assume without loss of generality that $(z';z'';u';u'')$ is a normal feasible point. But then by Lemma 1.9 we must have that $H(z';z'';u';u'')$ is a normal C-preconditioner and

$$w(H(z';z'';u';u'')) = W_k(z';z'';u';u'') < W_k(H(Y_k)) = w(\boldsymbol{n}_k(Y_k)),$$

which contradicts the assumption that Y_k solves problem 4. Therefore, the range of $\tilde{S}$ is a subset of the set of solution equivalence classes of problem 6.

On the other hand, suppose that E is a solution equivalence class of problem 6, but the normal C-preconditioner $S(E)$ does not solve problem 4. Also, let $(z'; z''; u'; u'')$ be the unique normal feasible point in E. Then there exists a normal C-preconditioner Y_k such that $w(Y_k) < w(S(E)) = W_k(z'; z''; u'; u'')$. But then by Lemma 1.9 we must have that $H(Y_k)$ is a normal feasible point of problem 6 and

$$W_k(H(Y_k)) = w(Y_k) < w(S(E)) = W_k(z'; z''; u'; u'')$$

This contradicts the assumption that E is a solution equivalence class of problem 6 since, by Lemma 1.11, E does not contain $H(Y_k)$ and the minimum objective function value of problem 6 over E is attained at $(z'; z''; u'; u'')$. Therefore, the range of $\tilde{S}$ is a superset of the set of solution equivalence classes of problem 6.

Hence, the range of $\tilde{S}$ is the set of solution equivalence classes of problem 6. Also, since S was a bijection we must have that $\tilde{S}$ is injective. Hence, the restriction $\tilde{S}$ of S to the set of solutions of problem 4 is a bijective function from that set to the set of solution equivalence classes of problem 6.

1.6 Linear Programming Formulations for Computing Some Left-Optimal and Right-Optimal C-Preconditioners

By Lemma 1.1, finding a (normal) preconditioner Y_k involves solving the nonlinear optimization problem

$$\max_{\underline{d}_k=1} \underline{x_k - \boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)}.$$

However, since x_k is fixed, we may solve

$$\min_{\underline{d}_k=1} \overline{\boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)}. \tag{7}$$

for Y_k to find a (normal) preconditioner. Similarly, finding a (normal) preconditioner Y_k involves solving the nonlinear optimization problem

$$\min_{\underline{d}_k=1} \overline{x_k - \boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)},$$

but we may solve

$$\max_{\underline{d}_k=1} \underline{\boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)}. \tag{8}$$

for Y_k to find a (normal) preconditioner.

In dealing with preconditioners, we applied Theorem 1.4 and replaced the nonlinear optimization problem 3 with the more tractable problem 4. The following theorem, analogous to Theorem 1.4, allows us to do a similar substitution when dealing with preconditioners and preconditioners. Given system 1, suppose that Y_k is a normal C-preconditioner and $\tilde{\boldsymbol{x}}_k$ is computed by equation 2. Then the following hold (in exact arithmetic):

1. If $0 \le \overline{n}_k$, then $\overline{\boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)} = \overline{n}_k$.

2. If $0 \ge \underline{n}_k$, then $\underline{\boldsymbol{n}_k(Y_k)/\boldsymbol{d}_k(Y_k)} = \underline{n}_k$.

3. If $\overline{n}_k < 0$, then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < \min\{-\underline{n}_k, ({}_k - x_k)\}$.

4. If $0 < \underline{n}_k$, then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < \min\{\overline{n}_k, (x_k - {}_k)\}$.

In particular, in the latter two cases, if $x_k = m(\boldsymbol{x}_k)$ then $w(\boldsymbol{x}_k \cap \tilde{\boldsymbol{x}}_k) < w(\boldsymbol{x}_k)/2$.

Hence, a normal C-preconditioner Y_k which solves

$$\min_{\underline{d}_k = 1} \overline{n}_k \tag{9}$$

is a normal preconditioner if $0 \le \overline{n}_k$, and a normal C-preconditioner Y_k which solves

$$\min_{\underline{d}_k = 1} -\underline{n}_k \tag{10}$$

is a normal preconditioner if $\underline{n}_k \le 0$. Otherwise, when $x_k = m(\boldsymbol{x}_k)$ at least half of $\boldsymbol{x}_k$ will be eliminated after intersection with $\tilde{\boldsymbol{x}}_k$. In addition, as we shall show, problems 9 and 10 can be stated as linear programming problems. These facts make solving problems 9 and 10 an attractive alternative to solving problems 7 and 8 when attempting to find preconditioners and preconditioners respectively.

Lemma 1.12 *Let Y_k be a C-preconditioner and suppose that for all $1 \le i \le n$ we have $x_i \in \{{}_i, m(\boldsymbol{x}_i), {}_i\}$. Define the following:*

$$\begin{aligned}
\mathcal{V}_M &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = m(\boldsymbol{x}_j) \text{ and } w(\boldsymbol{x}_j) \ne 0\}.\\
\mathcal{V}_L &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}.\\
\mathcal{V}_R &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}.
\end{aligned}$$

Also, choose $\delta_j \in [0,1]$ *for all* $j \in \mathcal{V}_M$*. Then for* $\mathcal{V}_I = \mathcal{V}_L$ *and* $\mathcal{V}_S = \mathcal{V}_R$ *we have*

$$\begin{aligned}
\underline{n}_k = \; & \sum_{i=1}^{m}(y_{k,i} - y_{k,i})m(\boldsymbol{b}_i) - \frac{1}{2}\sum_{i=1}^{m}(y_{k,i} + y_{k,i})w(\boldsymbol{b}_i) \\
& - \frac{1}{2}\sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)\delta_j \left[v_j - \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
& - \frac{1}{2}\sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)(1-\delta_j) \left[v_j + \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
& - \sum_{j \in \mathcal{V}_I} w(\boldsymbol{x}_j)v_j - \sum_{j \in \mathcal{V}_S} w(\boldsymbol{x}_j)v_j
\end{aligned}$$

and for $\mathcal{V}_I = \mathcal{V}_R$ *and* $\mathcal{V}_S = \mathcal{V}_L$ *we have*

$$\begin{aligned}
\overline{n}_k = \; & \sum_{i=1}^{m}(y_{k,i} - y_{k,i})m(\boldsymbol{b}_i) + \frac{1}{2}\sum_{i=1}^{m}(y_{k,i} + y_{k,i})w(\boldsymbol{b}_i) \\
& + \frac{1}{2}\sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)\delta_j \left[v_j - \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
& + \frac{1}{2}\sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)(1-\delta_j) \left[v_j + \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right) \right] \\
& + \sum_{j \in \mathcal{V}_S} w(\boldsymbol{x}_j)v_j + \sum_{j \in \mathcal{V}_I} w(\boldsymbol{x}_j)v_j
\end{aligned}$$

where

$$\begin{aligned}
v_j &= \sum_{i=1}^{m}(y_{k,i} - y_{k,i})({}_{i,j} + {}_{i,j}), \quad && j \in \mathcal{V}_M, \\
v_j &= \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), && j \in \mathcal{V}_I, \\
v_j &= \sum_{i=1}^{m}\left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), && j \in \mathcal{V}_S.
\end{aligned}$$

Based on these results, we may construct linear programming problems related to problems 9 and 10 when $X \in \boldsymbol{X}$ is chosen correctly. For $\boldsymbol{A} \in \mathbb{IR}m \times n$, these problems contains at most $2(n+m-1)$ variables and n constraints when in standard form.

Choose $x_i \in \{_i, m(\boldsymbol{x}_i), _i\}$ for all $1 \le i \le n$ and define the following:

$$\begin{aligned}
\mathcal{V}_M &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = m(\boldsymbol{x}_j) \text{ and } w(\boldsymbol{x}_j) \ne 0\}. \\
\mathcal{V}_L &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}. \\
\mathcal{V}_R &= \{j \mid 1 \le j \le n,\ \ j \ne k,\ \ x_j = {}_j \text{ and } w(\boldsymbol{x}_j) \ne 0\}.
\end{aligned}$$

Also, choose $\delta_j \in [0,1]$ for all $j \in \mathcal{V}_M$ and define the set of constraints

$$\begin{aligned}
v_j'' - v_j' + \sum_{i=1}^{m} (y_i' - y_i'')({}_{i,j} + {}_{i,j}) &= 0, \quad j \in \mathcal{V}_M, \\
v_j'' - v_j' + \sum_{i=1}^{m} \left(y_{k,i}'{}_{i,j} - y_{k,i}''{}_{i,j}\right) &= 0, \quad j \in \mathcal{V}_I, \\
v_j'' - v_j' + \sum_{i=1}^{m} \left(y_{k,i}'{}_{i,j} - y_{k,i}''{}_{i,j}\right) &= 0, \quad j \in \mathcal{V}_S, \qquad (11) \\
\sum_{i=1}^{m} (y_i'{}_{i,k} - y_i''{}_{i,k}) &= 1,
\end{aligned}$$

and

$$\begin{aligned}
&v_j' \ge 0, \quad v_j'' \ge 0, \quad j \in \mathcal{V}_M \cup \mathcal{V}_I \cup \mathcal{V}_S, \\
&y_i' \ge 0, \quad y_i'' \ge 0, \quad 1 \le i \le n,
\end{aligned}$$

where $\mathcal{V}_I$ and $\mathcal{V}_S$ will be set by each problem. Then we may state a linear programming problem corresponding to problem 9 as

minimize $\quad L_k(y'; y''; v'; v'') =$

$$\begin{aligned}
&- \sum_{i=1}^{m} (y_i' - y_i'') m(\boldsymbol{b}_i) + \frac{1}{2} \sum_{i=1}^{m} (y_i' + y_i'') w(\boldsymbol{b}_i) \\
&+ \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) \delta_j \left[v_j' - \sum_{i=1}^{m} (y_i'{}_{i,j} - y_i''{}_{i,j}) \right] \\
&+ \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)(1 - \delta_j) \left[v_j'' + \sum_{i=1}^{m} (y_i'{}_{i,j} - y_i''{}_{i,j}) \right] \qquad (12) \\
&+ \sum_{j \in \mathcal{V}_S} w(\boldsymbol{x}_j) v_j' + \sum_{j \in \mathcal{V}_I} w(\boldsymbol{x}_j) v_j''
\end{aligned}$$

subject to $\quad$ the set of constraints 11, where $\mathcal{V}_I = \mathcal{V}_R$ and $\mathcal{V}_S = \mathcal{V}_L$.

Similarly we may state a linear programming problem corresponding to problem 10 as

$$
\begin{aligned}
\text{minimize} \quad & R_k(y';y'';v';v'') = \\
& \sum_{i=1}^{m}(y'_i - y''_i)m(\boldsymbol{b}_i) + \frac{1}{2}\sum_{i=1}^{m}(y'_i + y''_i)w(\boldsymbol{b}_i) \\
& + \frac{1}{2}\sum_{j\in\mathcal{V}_M} w(\boldsymbol{x}_j)\delta_j \left[v'_j - \sum_{i=1}^{m}(y'_{i}{}_{i,j} - y''_i{}_{i,j})\right] \\
& + \frac{1}{2}\sum_{j\in\mathcal{V}_M} w(\boldsymbol{x}_j)(1-\delta_j) \left[v''_j + \sum_{i=1}^{m}(y'_{i}{}_{i,j} - y''_i{}_{i,j})\right] \\
& + \sum_{j\in\mathcal{V}_I} w(\boldsymbol{x}_j)v''_j + \sum_{j\in\mathcal{V}_S} w(\boldsymbol{x}_j)v'_j \\
\text{subject to} \quad & \text{the set of constraints 11, where} \\
& \mathcal{V}_I = \mathcal{V}_L \text{ and } \mathcal{V}_S = \mathcal{V}_R.
\end{aligned}
\tag{13}
$$

After solving problem 12 or 13 for $(y';y'';v';v'')$, a preconditioner Y_k is obtained as follows:

Definition 1.10 *If $(y';y'';v';v'')$ is a feasible point of problem 12 or 13, we define the preconditioner generated by $(y';y'';v';v'')$ to be $Y_k = H(y';y'';v';v'')$, which is computed by setting $y_{k,i}y'_i - y''_i$ for $1 \le i \le n$.*

As in the analysis of problem 6, we define normal feasible points as

Definition 1.11 *A feasible point $(y';y'';v';v'')$ of problem 12 or 13 is* <u>*normal*</u> *iff $y'_i y''_i = 0$ for all $1 \le i \le n$ and $v'_j v''_j = 0$ for all $j \in \mathcal{V}_M \cup \mathcal{V}_I \cup \mathcal{V}_S$.*

Also, as is the case for problem 6, normal feasible points have the following property.

Lemma 1.13 *No two distinct normal feasible points of problem 12 or 13 can generate equivalent preconditioners.*

We now show that preconditioners generated by feasible points of problems 12 and 13 must be C-preconditioners. For the proof, see the proof of Lemma 1.7.

Lemma 1.14 *Let $(y';y'';v';v'')$ be a feasible point of problem 12 or 13 and let Y_k be its generated preconditioner. Then Y_k is a C-preconditioner with $\underline{d}_k \geq 1$. Furthermore, if $(y';y'';v';v'')$ is a normal feasible point, then Y_k is a normal C-preconditioner ($\underline{d}_k = 1$).*

The above results allow us to give the following characterization of the relationships between problems 9 and 12, and between problems 10 and 13, for appropriate choices of X when only *normal* feasible points of the linear programming problems are considered.

Lemma 1.15 *If X is chosen as specified in the definition of problem 12, then H is a bijective function from the set of normal feasible points of problem 12 to the set of normal C-preconditioners such that for any normal feasible point $(y';y'';v';v'')$ and any normal C-preconditioner Y_k we have*

$$Y_k = H(y';y'';v';v'') \Rightarrow L_k(y';y'';v';v'') = \overline{n}_k.$$

Similarly, if X is chosen as specified in the definition of problem 13, then H is a bijective function from the set of normal feasible points of problem 13 to the set of normal C-preconditioners such that for any normal feasible point $(y';y'';v';v'')$ and any normal C-preconditioner Y_k we have

$$Y_k = H(y';y'';v';v'') \Rightarrow R_k(y';y'';v';v'') = -\underline{n}_k.$$

Furthermore, in both of the above cases, $H(Y_k) = (Y_k;Y_k;v;v)$ where

$$\begin{aligned} v_j &= \sum_{i=1}^{m} y_{k,i}({}_{i,j} + {}_{i,j}), & j \in \mathcal{V}_M, \\ v_j &= \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), & j \in \mathcal{V}_I, \\ v_j &= \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), & j \in \mathcal{V}_S. \end{aligned}$$

By Lemma 1.12 and Corollary 1.3, every normal C-preconditioner Y_k generates a normal feasible point $H(Y_k) = (y';y'';v';v'')$ of problem 12 for which we have $L_k(y';y'';v';v'') = \overline{n}_k$. Similarly, every normal C-preconditioner Y_k generates a normal feasible point $H(Y_k) = (y';y'';v';v'')$ of problem 13 for which we have $R_k(y';y'';v';v'') = -\underline{n}_k$. The relation $H(H(Y_k)) = Y_k$ follows easily from the definitions of H and H.

Conversely, by Lemma 1.14 every normal feasible point $(y';y'';v';v'')$ of problem 12 or 13 generates a normal C-preconditioner $H(y';y'';v';v'') = Y_k$. By Lemma 1.13 and the preceding case in this proof we must have that $H(H(y';y'';v';v'')) = (y';y'';v';v'')$.

We now state an analogue to Lemma 1.10 for problems 12 and 13 which allows us to effectively disregard non-normal feasible points. Note however that, unlike the case for problem 6, it is necessary to impose restrictions the objective functions.

Lemma 1.16 *Let $(y';y'';v';v'')$ be a feasible point of problem 12. Then we may construct a normal feasible point $(z';z'';u';u'')$ associated with $(y';y'';v';v'')$ such that $H(z';z'';u';u'')$ is a normal C-preconditioner equivalent to the C-preconditioner $H(y';y'';v';v'')$. Furthermore, we have*

$$L_k(z';z'';u';u'')\left\{\begin{array}{llrcl} \leq & L_k(y';y'';v';v'') & \textit{if} \quad L_k(y';y'';v';v'') & \geq & 0, \\ < & 0 & \textit{if} \quad L_k(y';y'';v';v'') & < & 0. \end{array}\right.$$

Similarly, let $(y';y'';v';v'')$ be a feasible point of problem 13. Then we may construct a normal feasible point $(z';z'';u';u'')$ associated with $(y';y'';v';v'')$ such that $H(z';z'';u';u'')$ is a normal C-preconditioner equivalent to the C-preconditioner $H(y';y'';v';v'')$. Furthermore, we have

$$R_k(z';z'';u';u'')\left\{\begin{array}{llrcl} \leq & R_k(y';y'';v';v'') & \textit{if} \quad R_k(y';y'';v';v'') & \geq & 0, \\ < & 0 & \textit{if} \quad R_k(y';y'';v';v'') & < & 0. \end{array}\right.$$

Let Y_k be the C-preconditioner generated by $(y';y'';v';v'')$ and define

$$v_j = \sum_{i=1}^{m} (y_{k,i} - y_{k,i})({}_{i,j} + {}_{i,j}), \quad j \in \mathcal{V}_M,$$

$$v_j = \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), \qquad j \in \mathcal{V}_I.$$

$$v_j = \sum_{i=1}^{m} \left(y_{k,i_{i,j}} - y_{k,i_{i,j}} \right), \qquad j \in \mathcal{V}_S.$$

Also, define $\alpha_i = \min\{y_i', y_i''\} \geq 0$ for $1 \leq i \leq n$, and $\beta_j = \min\{v_j', v_j''\} \geq 0$ for $j \in \mathcal{V}_M$. Then we have $y_{k,i} = y_i' - \alpha_i$ and $y_{k,i} = y_i'' - \alpha_i$ for all $1 \leq i \leq n$, as well as $v_j = v_j' - \beta_j$ and $v_j = v_j'' - \beta_j$ for all $j \in \mathcal{V}_M$.

Now, for all $j \in \mathcal{V}_I$ we have

$$\begin{aligned} v_j - v_j &= \sum_{i=1}^{m} \Big(y_{k,i_{i,j}} - y_{k,i_{i,j}} \Big) \\ &= \sum_{i=1}^{m} \big(y'_{k,ii,j} - y''_{k,ii,j} \big) + \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}) \\ &= v'_j - v''_j + \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}). \end{aligned}$$

Since $v'_j \geq 0$ and $v''_j \geq 0$ we must have $v_j \leq v''_j$ if $v_j = 0$. If $v_j \neq 0$ then we must have $v_j = 0$ from which it follows that

$$v_j = v''_j - v'_j - \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}) \leq v''_j.$$

Hence, for all $j \in \mathcal{V}_I$ there exists a $\beta_j \geq 0$ such that $v_j = v''_j - \beta_j$. Similarly, for all $j \in \mathcal{V}_S$ we have

$$\begin{aligned} v_j - v_j &= \sum_{i=1}^{m} \Big(y_{k,i_{i,j}} - y_{k,i_{i,j}} \Big) \\ &= \sum_{i=1}^{m} \big(y'_{k,ii,j} - y''_{k,ii,j} \big) - \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}) \\ &= v'_j - v''_j - \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}). \end{aligned}$$

Since $v'_j \geq 0$ and $v''_j \geq 0$ we must have $v_j \leq v'_j$ if $v_j = 0$. If $v_j \neq 0$ then we must have $v_j = 0$ from which it follows that

$$v_j = v'_j - v''_j - \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}) \leq v'_j.$$

Hence, for all $j \in \mathcal{V}_S$ there exists a $\beta_j \geq 0$ such that $v_j = v'_j - \beta_j$.

Then by Lemma 1.12 we must have

$$\overline{n}_k = \sum_{i=1}^{m} (y_{k,i} - y_{k,i}) m(\boldsymbol{b}_i) + \frac{1}{2} \sum_{i=1}^{m} (y_{k,i} + y_{k,i}) w(\boldsymbol{b}_i)$$

$$
\begin{aligned}
& + \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) \delta_j \left[v_j - \sum_{i=1}^{m} \Big(y_{k,i_{i,j}} - y_{k,i_{i,j}} \Big) \right] \\
& + \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)(1 - \delta_j) \left[v_j + \sum_{i=1}^{m} \Big(y_{k,i_{i,j}} - y_{k,i_{i,j}} \Big) \right] \\
& + \sum_{j \in \mathcal{V}_S} w(\boldsymbol{x}_j) v_j + \sum_{j \in \mathcal{V}_I} w(\boldsymbol{x}_j) v_j \\
= \quad & \sum_{i=1}^{m} (y_i' - y_i'') w(\boldsymbol{b}_i) + \frac{1}{2} \sum_{i=1}^{m} (y_i' + y_i'') w(\boldsymbol{b}_i) \\
& + \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) \delta_j \left[v_j' - \sum_{i=1}^{m} (y_i'{}_{i,j} - y_i''{}_{i,j}) \right] \\
& + \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j)(1 - \delta_j) \left[v_j'' + \sum_{i=1}^{m} (y_i'{}_{i,j} - y_i''{}_{i,j}) \right] \\
& - \left\{ \sum_{i=1}^{m} \alpha_i w(\boldsymbol{b}_i) + \frac{1}{2} \sum_{j \in \mathcal{V}_M} w(\boldsymbol{x}_j) \left[\beta_j + \sum_{i=1}^{m} \alpha_i w(\boldsymbol{a}_{i,j}) \right] \right\} \\
& + \sum_{j \in \mathcal{V}_S} w(\boldsymbol{x}_j) v_j' + \sum_{j \in \mathcal{V}_I} w(\boldsymbol{x}_j) v_j'' \\
& - \left\{ \sum_{j \in \mathcal{V}_S \cup \mathcal{V}_I} w(\boldsymbol{x}_j) \beta_j \right\} \\
\leq \quad & L_k(y'; y''; v'; v'').
\end{aligned}
$$

Also, by an almost identical arguement, we must have

$$-\underline{n}_k \leq R_k(y'; y''; v'; v'').$$

By Lemma 1.14, $\underline{d}_k \geq 1$. If we define $Z_k = \frac{1}{\underline{d}_k} Y_k$, then Z_k is a normal C-preconditioner equivalent to Y_k.

If $(y'; y''; v'; v'')$ is a feasible point of problem 12 then by Lemma 1.15, the point $(z'; z''; u'; u'') = H(Z_k)$ is a normal feasible point of problem 12 for which

$$\begin{aligned} L_k(z';z'';u';u'') &= \overline{n}_k(Z_k) \\ &= \frac{1}{\underline{d}_k}\overline{n}_k \\ &\leq \frac{1}{\underline{d}_k}L_k(y';y'';v';v''). \end{aligned}$$

Hence we have

$$L_k(z';z'';u';u'')\begin{cases} \leq & L_k(y';y'';v';v'') \quad \text{if} \quad L_k(y';y'';v';v'') \geq 0, \\ < & 0 \quad \text{if} \quad L_k(y';y'';v';v'') < 0. \end{cases}$$

Similarly, if $(y';y'';v';v'')$ is a feasible point of problem 13 then by Lemma 1.15, the point $(z';z'';u';u'') = H(Z_k)$ is a normal feasible point of problem 13 for which

$$\begin{aligned} R_k(z';z'';u';u'') &= -\underline{n}_k(Z_k) \\ &= -\frac{1}{\underline{d}_k}\underline{n}_k \\ &\leq \frac{1}{\underline{d}_k}R_k(y';y'';v';v''). \end{aligned}$$

$$R_k(z';z'';u';u'')\begin{cases} \leq & R_k(y';y'';v';v'') \quad \text{if} \quad R_k(y';y'';v';v'') \geq 0, \\ < & 0 \quad \text{if} \quad R_k(y';y'';v';v'') < 0. \end{cases}$$

Finally, by Lemma 1.15, the normal C-preconditioner generated by $(z';z'';u';u'')$ is in fact Z_k.

As was the case for problem 6, we are not interested in the feasible points of the linear programming problems themselves. We are interested in the generated preconditioners. Hence we make the following definitions.

Definition 1.12 *Two feasible points of problem 12 or problem 13 are* <u>*equivalent*</u> *iff they generate equivalent preconditioners. An equivalence class containing a solution to problem 12 or problem 13 is called a* <u>*solution equivalence class*</u>*.*

By Lemma 1.13 and 1.16, we have the following result.

Lemma 1.17 *Each equivalence class of feasible points contains a unique normal feasible point. Furthermore, if the objective function is nonnegative over an entire class, then the minimum objective function value of problem 6 over that class is attained at the normal feasible point in that class. Finally, if the value of the objective function at a normal feasible point is nonnegative, then it is nonnegative for any other feasible point in the same class.*

We are now able to prove the following theorem relating problems 9 and 12, or problems 10 and 13, for appropriate choices of X when the objective function of problem 9, or problem 10, is nonnegative for all C-preconditioners. If $x_i \in \{{}_i, m(\boldsymbol{x}_i), {}_i\}$ for all $1 \le i \le n$ and if $\overline{n}_k \ge 0$ for all (normal) C-preconditioners, then problem 9 and problem 12 are equivalent in the following sense:

1. Each of the problems is feasible iff a C-preconditioner exists.
2. There is a bijective function S from the set of normal C-preconditioners (feasible points of problem 9) to the set of equivalence classes of feasible points of problem 12 such that for any normal C-preconditioner Y_k and for any $(z'; z''; u'; u'') \in S(Y_k)$ we have
$$\overline{n}_k = L_k(y'; y''; v'; v'') \le L_k(z'; z''; u'; u''),$$
where $(y'; y''; v'; v'')$ is the unique normal feasible point in $S(Y_k)$.
3. The restriction $\tilde{S}$ of S to the set of solutions of problem 9 is a bijective function from that set to the set of solution equivalence classes of problem 12.

Similarly, if $x_i \in \{{}_i, m(\boldsymbol{x}_i), {}_i\}$ for all $1 \le i \le n$ and if $-\underline{n}_k \ge 0$ for all (normal) C-preconditioners, then problem 10 and problem 13 are equivalent in the following sense:

1. Each of the problems is feasible iff a C-preconditioner exists.
2. There is a bijective function S from the set of normal C-preconditioners (feasible points of problem 10) to the set of equivalence classes of feasible points of problem 13 such that for any normal C-preconditioner Y_k and for any $(z'; z''; u'; u'') \in S(Y_k)$ we have
$$-\underline{n}_k = R_k(y'; y''; v'; v'') \le R_k(z'; z''; u'; u''),$$
where $(y'; y''; v'; v'')$ is the unique normal feasible point in $S(Y_k)$.

3. The restriction $\tilde{S}$ of S to the set of solutions of problem 10 is a bijective function from that set to the set of solution equivalence classes of problem 13.

Here we only give the proof of the first half, which relates problems 9 and 12, as the proof of the second half, which relates problems 10 and 13, is almost identical.

Recall that existence of a C-preconditioner implies existence of an equivalent normal C-preconditioner. Then clearly problem 9 is feasible iff a C-preconditioner exists, as it is a minimization problem over all normal C-preconditioners. Also, existence of a normal C-preconditioner implies existence of a (normal) feasible point for problem 12 by Lemma 1.12. Conversely, every feasible point of problem 12 generates a C-preconditioner by Lemma 1.14. Hence the first statement is proved.

Next, recall that the function H defined in Lemma 1.15 is a bijection between the set of normal C-preconditioners and the set of normal feasible points. Since each equivalence class of feasible points contains a unique normal feasible point by Lemma 1.17, we may define $S(Y_k)$ to be the equivalence class of feasible points containing $H(Y_k)$. Similarly, for any equivalence class E of feasible points of problem 12 we may define $S(E)$ to be the normal C-preconditioner $H(y';y'';v';v'')$, where $(y';y'';v';v'')$ is the unique normal feasible point contained in E. It follows that S must be bijective with inverse S. Then by Lemma 1.15, Lemma 1.16, and Lemma 1.17 we must have $\overline{n}_k = L_k(y';y'';v';v'') \le L_k(z';z'';u';u'')$, where Y_k is a normal C-preconditioner, $(y';y'';v';v'')$ is the unique normal feasible point in $S(Y_k)$, and $(z';z'';u';u'') \in S(Y_k)$ is arbitrary. This proves the second statement.

Now, let Y_k be a normal C-preconditioner which solves problem 9 and suppose that $S(Y_k)$ is not a solution equivalence class of problem 12. By Lemma 1.15, $H(Y_k)$ is a normal feasible point of problem 12 for which we have $\overline{N}_k(\boldsymbol{n}_k(Y_k)) = L_k(H(Y_k))$. Since $S(Y_k)$ was not a solution equivalence class, there must exist a feasible point $(z';z'';u';u'')$ of problem 12 for which we have $L_k(z';z'';u';u'') < L_k(H(Y_k))$. By Lemma 1.16 we may assume without loss of generality that $(z';z'';u';u'')$ is a normal feasible point. But then by Lemma 1.15 we must have that $H(z';z'';u';u'')$ is a normal C-preconditioner and

$$\overline{N}_k(H(z';z'';u';u'')) = L_k(z';z'';u';u'') < L_k(H(Y_k)) = \overline{N}_k(\boldsymbol{n}_k(Y_k)),$$

which contradicts the assumption that Y_k solves problem 9. Therefore, the range of $\tilde{S}$ is a subset of the set of solution equivalence classes of problem 12.

On the other hand, suppose that E is a solution equivalence class of problem 12, but the normal C-preconditioner $S(E)$ does not solve problem 9. Then there exists a normal C-preconditioner Y_k such that $\overline{N}_k(Y_k) < \overline{N}_k(S(E)) = L_k(z'; z''; u'; u'')$, where $(z'; z''; u'; u'')$ is the unique normal feasible point in E. But then by Lemma 1.15 we must have that $H(Y_k)$ is a normal feasible point of problem 12 and

$$L_k(H(Y_k)) = \overline{N}_k(Y_k) < \overline{N}_k(S(E)) = L_k(z'; z''; u'; u'')$$

This contradicts the assumption that E is a solution equivalence class of problem 12 since, by Lemma 1.17, E does not contain $H(Y_k)$ and the minimum objective function value of problem 12 over E is attained at $(z'; z''; u'; u'')$. Therefore, the range of $\tilde{S}$ is a superset of the set of solution equivalence classes of problem 12.

Hence, the range of $\tilde{S}$ is the set of solution equivalence classes of problem 12. Also, since S was a bijection we must have that $\tilde{S}$ is injective. Hence, the restriction $\tilde{S}$ of S to the set of solutions of problem 9 is a bijective function from that set to the set of solution equivalence classes of problem 12.