Wir wenden uns nun der Initialisierung zu. Die Prozedur InitSimplex transformiert ein gegebenes lineares Programm in Standardform in ein Äquivalentes in Schlupfform dessen Basislösung zulässig ist oder liefert "unlösbar" zurück falls das Eingabeproblem unlösbar ist. Zunächst kann man leicht beobachten, dass die Basislösung einer Schlupfform zulässig ist genau dann wenn $b \geq 0$ ist. Für die Initialisierung des Simplex-Verfahrens (Phase 1) ist also vor allem der Fall wo $b \not\geq 0$ ist interessant. In diesem Fall gehen wir wie folgt vor:

Definition 10.10. Sei $L = A, b, c$ ein lineares Programm in Standardform mit $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ und $c \in \mathbb{R}^n$ so dass $b \not\geq 0$. Sei $k \in \{1, \ldots, m\}$ so dass b_k minimal ist. Dann definieren wir das lineare Programm S_L in Schlupfform in den Variablen $\{x_0, \ldots, x_{m+n}\}$ mit $F = \{1, \ldots, n, n+k\}$ und $S = \{0, n+1, \ldots, n+k-1, n+k+1, \ldots, n+m\}$ als:

$$
\begin{aligned}
z &= b_k - \sum_{j=1}^{n} a_{k,j} x_j - x_{n+k} \max! \\
x_0 &= -b_k + \sum_{j=1}^{n} a_{k,j} x_j + x_{n+k} \\
x_{n+i} &= (b_i - b_k) + \sum_{j=1}^{n} (a_{k,j} - a_{i,j}) x_j + x_{n+k} \quad \text{für } i \in \{1, \ldots, k-1, k+1, \ldots, m\}
\end{aligned}
$$

Lemma 10.5. *Sei $L = A, b, c$ ein lineares Programm mit $b \not\geq 0$, dann ist die Basislösung von S_L zulässig. Weiters ist L lösbar genau dann wenn S_L eine optimale Lösung hat mit $x_0 = 0$.*

Beweis. Zunächst beobachten wir dass die Basislösung von S_L zulässig ist, da $-b_k \geq 0$ und, für alle $i \in \{1, \ldots, m\}$ gilt dass $b_i \geq b_k$ und damit $b_i - b_k \geq 0$.

Um die zweite Behauptung zu zeigen, transformieren wir zunächst $L =$

$$
\begin{aligned}
z &= \sum_{j=1}^{n} c_j x_j \max! \\
\sum_{j=1}^{n} a_{i,j} x_j &\leq b_i \quad \text{für } i = 1, \ldots, m \\
x_j &\geq 0 \quad \text{für } j = 1, \ldots, n
\end{aligned}
$$

in das folgende Hilfsprogramm L' in Standardform:

$$
\begin{aligned}
z &= -x_0 \max! \\
\sum_{j=1}^{n} a_{i,j} x_j - x_0 &\leq b_i \quad \text{für } i = 1, \ldots, m \\
x_j &\geq 0 \quad \text{für } j = 0, \ldots, n
\end{aligned}
$$

Zunächst sieht man dass L' lösbar ist: für beliebige $x_1, \ldots, x_n \geq 0$ wählen wir einfach x_0 hinreichend groß um alle Ungleichungen zu erfüllen. Weiters gilt: L ist lösbar genau dann wenn L' eine optimale Lösung mit $x_0 = 0$ hat. Falls nämlich L lösbar ist, dann erhält man durch Setzung von $x_0 = 0$ eine Lösung von L' und diese Lösung ist optimal da ja $x_0 \geq 0$ sein muss. Für die Gegenrichtung sei $(0, x_1, \ldots, x_n)$ optimale Lösung von L', dann ist $(x_1, \ldots, x_n)$ auch Lösung von L.

Die Schlupfform von L' ist:

$$z = -x_0 \max!$$
$$x_{n+i} = b_i - \sum_{j=1}^{n} a_{i,j} x_j + x_0 \quad \text{für } i = 1, \ldots, m$$
$$x_j \geq 0 \quad \text{für } j = 0, \ldots, n+m$$

mit $F = \{0, 1, \ldots, n\}$ und $S = \{n+1, \ldots, n+m\}$. Ein Austausch von x_0 mit x_{n+k} führt dann über die Gleichung $x_0 = -b_k + \sum_{j=1}^{n} a_{k,j} x_j + x_{n+k}$ zu S_L. □

Auf Basis dieses Resultats kann jetzt die Prozedur InitSimplex(A, b, c) wie in Algorithmus 44 angegeben werden, wobei wir davon ausgehen dass die Prozedur Simplex-Schleife mit einer Schlupfform als Ergebnis antwortet. Damit ist das Simplex-Verfahren vollständig spezifiziert

Algorithmus 44 Initialisierung des Simplex-Algorithmus

```
Prozedur InitSimplex(A, b, c)
    Falls b ≥ 0 dann
        Antworte Schlupfform(A, b, c)
    sonst
        U := Simplex-Schleife(S_{A,b,c})
        (x_0, ..., x_{n+m}) := Basislösung(U)
        Falls x_0 = 0 dann
            U' := Transformiere U in Schlupfform von A, b, c
            Antworte U'
        sonst
            Antworte "unlösbar"
        Ende Falls
    Ende Falls
Ende Prozedur
```

und wir können nun seine partielle Korrektheit beweisen.

Satz 10.1. *Sei $L = A, b, c$ ein lineares Programm in Standardform so dass* Simplex(A, b, c) *terminiert, dann gilt:*

1. *Falls L unbeschränkt ist, dann antwortet* Simplex(A, b, c) *mit "unbeschränkt".*

2. *Falls L unlösbar ist, dann antwortet* Simplex(A, b, c) *mit "unlösbar".*

3. *Falls L eine optimale Lösung mit Zielwert $z \in \mathbb{R}$ hat, dann liefert* Simplex(A, b, c) *eine Lösung mit Zielwert z.*

Beweis. Da Simplex(A, b, c) terminiert, terminiert auch die Simplex-Schleife im Aufruf InitSimplex(A, b, c) mit, nach Lemma 10.4, der optimalen Lösung $x = (x_0, \ldots, x_{n+m})$ von S_L. Falls L unlösbar ist, dann ist nach Lemma 10.5 $x_0 \neq 0$ und damit antwortet die Prozedur InitSimplex(A, b, c) mit "unlösbar" und das Verfahren wird abgebrochen. Falls L lösbar ist, dann ist nach Lemma 10.5 $x_0 = 0$ und damit erzeugt die Prozedur InitSimplex(A, b, c) eine Schlupfform von A, b, c mit zulässiger Basislösung. Da Simplex(A, b, c) terminiert, terminiert nun auch die Simplex-Schleife und damit folgt dieser Satz unmittelbar aus Lemma 10.4. □

Offen ist jetzt noch die Termination des Simplex-Verfahrens. Dazu wollen wir uns zunächst überlegen wie es passieren kann, dass der Zielwert nicht strikt steigt. Da nur solche $f \in F$ ausgewählt werden die $c_f > 0$ haben kann dieser Fall nur eintreten wenn eine Gleichung für ein $s \in S$ gemeinsam mit ihrer Nichtnegativitätsbedingung $x_s \geq 0$ die Variable x_f so einschränkt dass x_f gar nicht erhöht werden kann. In diesem Fall muss $b_s = 0$ sein und wir sprechen von einer degenerierten Schlupfform.

Beispiel 10.7. Betrachten wir zum Beispiel

$$\begin{aligned} z &= 3 - \frac{1}{2}x_1 + 2x_2 \\ x_3 &= 1 - \frac{1}{2}x_1 \\ x_4 &= x_1 - x_2 \end{aligned}$$

Diese Schlupfform hat die zulässige Basislösung $(0, 0, 1, 0)$ mit Zielwert 3. Dann wird die freie Variable x_2 zum Vertauschen ausgewählt. Nun schränkt aber die Gleichung und Nichtnegativitätsbedingung von x_4 die Variable x_2 auf $x_2 \leq 0$ ein. Wir erhalten durch den Austausch die Schlupfform

$$\begin{aligned} z &= 3 + \frac{3}{2}x_1 - 2x_4 \\ x_2 &= x_1 - x_4 \\ x_3 &= 1 - \frac{1}{2}x_1 \end{aligned}$$

die ebenfalls die zulässige Basislösung $(0, 0, 1, 0)$ mit Zielwert 3 hat. In diesem Schritt hat sich der Zielwert also nicht verändert. Nun muss die freie Variable x_1 ausgewählt werden die durch die Gleichung von x_3 am stärksten eingeschränkt wird. Nach Durchführung dieser Vertauschung erhalten wir die Schlupfform

$$\begin{aligned} z &= 6 - 3x_3 - 2x_4 \\ x_1 &= 2 - 2x_3 \\ x_2 &= 2 - 2x_3 - x_4 \end{aligned}$$

mit Basislösung $(2, 2, 0, 0)$ und Zielwert 6 wobei es sich, da alle $c_j \leq 0$ sind, um eine optimale Lösung handelt.

Falls Zwischenschritte auftreten, in denen sich der Wert der Zielfunktion nicht verändert, spricht man von *stalling*. Das tritt in der Praxis häufig auf und ist (bis auf den Zeitaspekt) kein Problem solange der Algorithmus zu einem späteren Zeitpunkt wieder einen Schritt durchführt der den Zielwert strikt erhöht. Allerdings kann es auch dazu kommen, das der Algorithmus kreist, d.h. dass er von einer Schlupfform U nach einigen Schritten die den Zielwert nicht erhöhen wieder bei U ankommt. Dieses Verhalten, wenn auch theoretisch möglich, tritt in der Praxis so selten auf, dass die meisten Implementierungen es gar nicht behandeln.

Beispiel 10.8. Die Schlupfform

$$\begin{aligned} z &= 10x_1 - 57x_2 - 9x_3 - 24x_4 \\ x_5 &= -\frac{1}{2}x_1 + \frac{11}{2}x_2 + \frac{5}{2}x_3 - 9x_4 \\ x_6 &= -\frac{1}{2}x_1 + \frac{3}{2}x_2 + \frac{1}{2}x_3 - x_4 \\ x_7 &= 1 - x_1 \end{aligned}$$

kann mit den Austauschschritten $x_1 \leftrightarrow x_5$, $x_2 \leftrightarrow x_6$, $x_3 \leftrightarrow x_1$, $x_4 \leftrightarrow x_2$, $x_5 \leftrightarrow x_3$, $x_4 \leftrightarrow x_6$ wieder auf sich selbst zurückgeführt werden.

Es gibt verschiedene Strategien um Kreisen zu verhindern, z.B. die Regel von Bland die darin besteht bei mehreren Möglichkeiten für die Auswahl eines x_i immer jene zu bevorzugen wo i minimal ist. Mit einer derartigen Erweiterung kann die Termination des Simplex-Verfahren sichergestellt werden. Für dieses Verfahren gilt dann eine entsprechend stärkere Form von Satz 10.1.