Closure of Regular sets under Reversal

For a string $w$, let $w^R$ denote the reversal of $w$, and for a set $L$, let $\text{Rev}(L)$ denote the set \{w$^R$: $w \in L$\}. That regularity of $L$ implies regularity of $\text{Rev}(L)$ may be proved in at least three different ways:

1. Start with a DFA, say $M$, which accepts a regular language $L$, and build an NFA (with $\varepsilon$-transitions) to accept $\text{Rev}(L)$ by appropriately introducing a new start state, say $p$, with an $\varepsilon$-transition to each of the final states of $M$ (that are no longer final states of the new machine), reversing the arcs of $M$, and making the start state of $M$ the sole final state of the new machine.

2. Start with a right-linear grammar, say $G$, which generates a regular language $L$, and build a grammar by systematically reversing the right side of each production of $G$. The resulting grammar generates $\text{Rev}(L)$ and is necessarily left-linear, hence $\text{Rev}(L)$ is regular. (Equivalently, start with a left-linear grammar and likewise build a right-linear grammar.)

3. Start with a regular expression, say $r$, for a regular language $L$, and build a regular expression for $\text{Rev}(L)$. To that end, proceed by induction on the number of operators in $r$.

The present note amplifies the third scheme outlined above. First a few lemmas.

**Lemma 1:** $\text{Rev}(L_1 \cup L_2) = \text{Rev}(L_1) \cup \text{Rev}(L_2)$.\[1\]

**Lemma 2:** $\text{Rev}(L_1 \cdot L_2) = \text{Rev}(L_2) \cdot \text{Rev}(L_1)$.

**Proof:** Let $w \in \text{Rev}(L_1 \cdot L_2)$. Then $w$ may be written as $w = (xy)^R$ where $x \in L_1$ and $y \in L_2$. Now, $(xy)^R = y^Rx^R$ that is clearly in $\text{Rev}(L_2) \cdot \text{Rev}(L_1)$. Accordingly, $\text{Rev}(L_1 \cdot L_2) \subseteq \text{Rev}(L_2) \cdot \text{Rev}(L_1)$.

For the reverse inclusion, let $w \in \text{Rev}(L_2) \cdot \text{Rev}(L_1)$. Then $w$ may be written as $w = y^Rx^R$ where $y \in L_2$ and $x \in L_1$. Now, $y^Rx^R = (xy)^R$ that is clearly in $\text{Rev}(L_1 \cdot L_2)$. Accordingly, $\text{Rev}(L_2) \cdot \text{Rev}(L_1) \subseteq \text{Rev}(L_1 \cdot L_2)$.\[2\]
**Lemma 3:** $\text{Rev}(L^*) = (\text{Rev}(L))^*$.

**Proof:** It is clear that the empty string $\varepsilon$ is in each of $\text{Rev}(L^*)$ and $(\text{Rev}(L))^*$. In what follows, all strings are of length at least one.

Let $w$ be a typical element of $\text{Rev}(L^*)$. Then $w = x^R$ for some $x \in L^*$. Note that $x$ may be written as $x = x_1 \ldots x_n$, where $n \geq 1$ and $x_i \in L$ for $1 \leq i \leq n$. Now, $w = x^R = (x_n \ldots x_1)^R = x_n^R \ldots x_1^R$. Since $x_i^R$ is in $\text{Rev}(L)$ for $1 \leq i \leq n$, it is clear that $x_n^R \ldots x_1^R$ (that is equal to $w$) is in $(\text{Rev}(L))^*$. Thus, $\text{Rev}(L^*) \subseteq (\text{Rev}(L))^*$.

For the reverse inclusion, let $w$ be a typical element of $(\text{Rev}(L))^*$. Then $w = w_1 \ldots w_n$, where $n \geq 1$ and $w_i \in \text{Rev}(L)$ for $1 \leq i \leq n$. This means that $w_i^R \in L$ for $1 \leq i \leq n$, i.e., $w_n^R \ldots w_1^R \in L^*$. Since $w_n^R \ldots w_1^R = (w_1 \ldots w_n)^R = w^R$, it is clear that $w^R$ is in $L^*$, and hence $(w^R)^R$ (that is equal to $w$) is in $\text{Rev}(L^*)$. Thus, $(\text{Rev}(L))^* \subseteq \text{Rev}(L^*)$.

---

**Theorem:** If $L$ is a regular set, then so is $\text{Rev}(L)$.

**Proof:** Let $L$ be a regular set. Accordingly, $L$ is denoted by a regular expression, say $r$. It suffices to show that $\text{Rev}(L)$ is denoted by a regular expression. Let $k$ be the number of operators in $r$. Induct on $k$ to prove the claim.

For $k = 0$, $r$ is of the form $\phi$, $\varepsilon$ or $a$ where $a$ is a member of the alphabet. Accordingly, $L$ is equal to one of $\emptyset$, $\{\varepsilon\}$ and $\{a\}$, whence $\text{Rev}(L) = L$, and the claim follows.

For $k \geq 1$, $r$ is of one of the following forms: $r_1 + r_2$, $r_1 \cdot r_2$ and $r_1^*$ where $r_1$ and $r_2$ are themselves regular expressions. Let $L_1$ and $L_2$ be the languages denoted by $r_1$ and $r_2$, respectively. It is clear that the number of operators in each of $r_1$ and $r_2$ is strictly less than $k$. By induction hypothesis, $\text{Rev}(L_1)$ and $\text{Rev}(L_2)$ are denoted by regular expressions, say $s_1$ and $s_2$, respectively.

(i) $r = r_1 + r_2$: In this case, $L = L_1 \cup L_2$. By Lemma 1, $\text{Rev}(L_1 \cup L_2) = \text{Rev}(L_1) \cup \text{Rev}(L_2)$ that is clearly denoted by the regular expression $s_1 + s_2$.

(ii) $r = r_1 \cdot r_2$: In this case, $L = L_1 \cdot L_2$. By Lemma 2, $\text{Rev}(L_1 \cdot L_2) = \text{Rev}(L_2) \cdot \text{Rev}(L_1)$ that is clearly denoted by the regular expression $s_2 \cdot s_1$.

(iii) $r = r_1^*$: In this case, $L = L_1^*$. By Lemma 3, $\text{Rev}(L_1^*) = (\text{Rev}(L_1))^*$ that is clearly denoted by the regular expression $s_1^*$.