1 Proof of Lemmas

Theorem 4.2. *Assume $score(u) \leq score(u')$ for any terms $u, u' \in \mathcal{T}_\Sigma(\mathcal{X})$ with $u \preccurlyeq_E u'$. Suppose that a term s is visited by Alg. 3 from an initial term t. If there exists a narrowing sequence (possibly not in the search graph by Alg 3):*

$$t = t_0 \rightsquigarrow_{\mathcal{R}} t_1 \rightsquigarrow_{\mathcal{R}} t_2 \rightsquigarrow_{\mathcal{R}} \cdots \rightsquigarrow_{\mathcal{R}} t_k$$

satisfying $score(t_i) < score(s)$ for $0 \leq i \leq k$, then there exists a term t'_k visited by the algorithm before s with $t_k \preccurlyeq_E t'_k$.

Proof. We prove this by applying induction on k. For $k = 0$, t is visited at the first iteration of the loop in line 4, For $k = N$, suppose that there exists a narrowing sequence: $t = t_0 \rightsquigarrow_{\mathcal{R}} t_1 \rightsquigarrow_{\mathcal{R}} \cdots \rightsquigarrow_{\mathcal{R}} t_k \rightsquigarrow_{\mathcal{R}} t_{k+1}$ By applying the induction hypothesis on the subsequence $t_0 \rightsquigarrow_{\mathcal{R}} \cdots \rightsquigarrow_{\mathcal{R}} t_k$, we get a term t'_k such that $t_k \preccurlyeq_E t'_k$ and t'_k is visited before s. Because $\preccurlyeq_E$ is a simulation with respect to $\rightsquigarrow_{\mathcal{R}}$, there exists a term t'_{k+1} such that $t'_k \rightsquigarrow_{\mathcal{R}} t'_{k+1}$ and $t_{k+1} \preccurlyeq_E t'_{k+1}$. Since we have assumed that the algorithm does not terminate before visiting s, t'_{k+1} is not a solution. There are two possibilities. If t'_{k+1} is subsumed by an already visited term t''_{k+1}, then we have $t_{k+1} \preccurlyeq_E t'_{k+1} \preccurlyeq_E t''_{k+1}$, and an already visited term t''_{k+1}, which completes the proof. Otherwise, t'_{k+1} is enqueued before s is dequeued. Since $score(t'_{k+1}) \leq score(t_{k+1}) < score(s)$, t'_{k+1} will be eventually dequeued before s. ◻

Lemma 4.3. *Let t be an initial term for Algorithm 3 If u is dequeued from the priority queue, then $t \rightsquigarrow^*_{\mathcal{R}} u$ holds.*

Proof. If s is dequeued from the priority queue, then s must have been enqueued either at line 3 or line 11.

We prove by complete induction on the number of iterations of the while loop starting at line 4 to show that if s is enqueued, then $t \rightsquigarrow^*_{\mathcal{R}} s$. If t is enqueued before the first iteration, at line 3, since $t \rightsquigarrow^*_{\mathcal{R}} t$, we are done. Suppose that s is enqueued at some later iteration. Then there must exists s', dequeued from the priority queue, such that $s' \rightsquigarrow_{\mathcal{R}} s$. s' must have been enqueued at some previous iteration, so we can apply the induction hypothesis to get $t \rightsquigarrow^*_{\mathcal{R}} s'$. From $t \rightsquigarrow^*_{\mathcal{R}} s'$ and $s' \rightsquigarrow_{\mathcal{R}} s$, we get $t \rightsquigarrow^*_{\mathcal{R}} s$. ◻