Supplementary Proofs for *Finding a Densest Segment*

Shin-Cheng Mu & Sharon Curtis

July 31, 2015

The proofs contained here are given in the same order as in the paper. For ease of reference, please note that equation/theorem references (1),(2)... from the paper are denoted (P.1), (P.2)... and those that occur only in the supplement are labelled (S.1), (S.2), etc.

Additional Definitions and Properties

Densities
For precise reference, we expand the density property of $\leq_d$ — equation (P.3) in the paper — into the following four properties labelled (S.$\leq_d$) to (S.$>_d$). For non-empty lists of elements x, y, we have:

(a)		(b)		(c)	
$x \leq_d x \mathbin{+\!\!+} y$	$\Leftrightarrow$	$x \mathbin{+\!\!+} y \leq_d y$	$\Leftrightarrow$	$x \leq_d y$	(S.$\leq_d$)
$x <_d x \mathbin{+\!\!+} y$	$\Leftrightarrow$	$x \mathbin{+\!\!+} y <_d y$	$\Leftrightarrow$	$x <_d y$	(S.$<_d$)
$x \geq_d x \mathbin{+\!\!+} y$	$\Leftrightarrow$	$x \mathbin{+\!\!+} y \geq_d y$	$\Leftrightarrow$	$x \geq_d y$	(S.$\geq_d$)
$x >_d x \mathbin{+\!\!+} y$	$\Leftrightarrow$	$x \mathbin{+\!\!+} y >_d y$	$\Leftrightarrow$	$x >_d y$	(S.$>_d$)

Maximum Operators
The binary operators $\upharpoonleft_{\trianglelefteq}$ and $\upharpoonright_{\trianglelefteq}$ (defined in Section 3.1 of the paper) are idempotent and associative. We will also need the following distributive properties. Firstly,

$$z \upharpoonleft_{\trianglelefteq} (x \upharpoonleft_{\trianglelefteq} y) \;=\; (z \upharpoonleft_{\trianglelefteq} x) \upharpoonleft_{\trianglelefteq} (z \upharpoonleft_{\trianglelefteq} y), \tag{S.1}$$

where x, y, z are non-empty lists of elements.

Also, $max_{\preceq}$ distributes over $(\mathbin{+\!\!+})$: that is,

$$max_{\preceq}\ (x \mathbin{+\!\!+} y) \;=\; max_{\preceq}\ x \ \upharpoonleft_{\preceq}\ max_{\preceq}\ y, \tag{S.2}$$

for any lists of elements x and y. This follows from the associativity of $\upharpoonleft_{\preceq}$.

Prefixes
Given the definition of *prefixes* as *inits*, we have that

$$\mathit{prefixes}\ (x + a : y)\ =\ \mathit{prefixes}\ x + \mathit{map}\ (x + [a]+)\ (\mathit{prefixes}\ y), \quad \text{(S.3)}$$

for any lists of elements x and y. This is easily proved by induction on x.

Also, for any lists of elements x and y and maximum operator $\uparrow_{\trianglelefteq}$, the following property holds:

$$\begin{array}{l} \mathit{max}_{\uparrow_{\trianglelefteq}}\,(\mathit{map}\,f\,(\mathit{prefixes}\,(x + y))) \quad = \\ \quad \mathit{max}_{\uparrow_{\trianglelefteq}}\,(\mathit{map}\,f\,(\mathit{prefixes}\,x)) \uparrow_{\trianglelefteq} \mathit{max}_{\uparrow_{\trianglelefteq}}\,(\mathit{map}\,(f \cdot (x+))\,(\mathit{prefixes}\,y)). \end{array} \quad \text{(S.4)}$$

This follows from the idempotence and associativity of $\uparrow_d$.

Proofs for Section 3

Proof of Lemma 3.1
First, we prove the implication $\mathit{width}\,x \leq L \ \Rightarrow\ x = \mathit{mdp}\ x$, which is carried out using induction on x:

Case [ ]

$$\begin{array}{ll} & \mathit{mdp}\ [\,] \\ = & \quad \{\text{definition of } \mathit{mdp} \text{ (P.10)}\} \\ & \mathit{max}_{\preceq}\ (\mathit{prefixes}\ [\,]) \\ = & \quad \{\text{definition of } \mathit{prefixes}\} \\ & \mathit{max}_{\preceq}\ [[\,]] \\ = & \quad \{\text{definition of } \mathit{max}\} \\ & [\,]. \end{array}$$

Case $x + [a]$

$$\begin{array}{ll} & \mathit{mdp}\ (x + [a]) \\ = & \quad \{\text{definition of } \mathit{mdp} \text{ (P.10)}\} \\ & \mathit{max}_{\preceq}\ (\mathit{prefixes}(x + [a])) \\ = & \quad \{\text{property of } \mathit{prefixes}\} \\ & \mathit{max}_{\preceq}\ (\mathit{prefixes}\ x\ +\ [x + [a]]) \\ = & \quad \{\text{property of } \mathit{max}\} \\ & \mathit{max}_{\preceq}\ (\mathit{prefixes}\ x)\ \uparrow_{\preceq}\ x + [a] \\ = & \quad \{\text{definition of } \mathit{mdp} \text{ (P.10)}\} \end{array}$$

$$\begin{aligned}
& \mathit{mdp}\ x \ \uparrow_{\preceq}\ x + [a] \\
= \quad & \{\text{induction, as from ((P.2)), } \mathit{width}\,(x + [a]) \leq L \Rightarrow\ \mathit{width}\,x \leq L\} \\
& x \ \uparrow_{\preceq}\ x + [a] \\
= \quad & \{\text{claim, see below}\} \\
& x + [a].
\end{aligned}$$

The claim above:
If $\mathit{width}\,(x + [a]) = L$, then from (P.2) we have $\mathit{width}\ x < \mathit{width}\,(x + [a])$ and (P.6) applies, so that $x \prec\ x + [a]$. Otherwise, if $\mathit{width}\,(x + [a]) < L$, then as we have $\mathit{width}\ x < \mathit{width}\,(x + [a])$ from (2), (P.8) applies, and we have $x \preceq\ x + [a]$, but not $x + [a] \preceq\ x$, so we have $x \prec\ x + [a]$.

Second, we prove that $\mathit{wide}\ x \Leftrightarrow \mathit{wide}\ (\mathit{mdp}\ x)$. Note that when $\mathit{width}\ x < L$, the above applies, and it immediately follows that $\mathit{width}\ (\mathit{mdp}\ x) < L$, because $x = \mathit{mdp}\ x$. So it suffices to show that when $\mathit{width}\ x \geq L$, it is also the case that $\mathit{width}\ (\mathit{mdp}\ x) \geq L$. In the following calculation, we define the predicate we as $\mathit{we}\ x = \mathit{width}\ x \geq L$ for convenience ("we" stands for wide-enough).

$$\begin{aligned}
& \mathit{width}\ (\mathit{mdp}\ x) \\
= \quad & \{\text{definition of } \mathit{mdp} \text{ (P.10)}\} \\
& \mathit{width}\ (\mathit{max}_{\preceq}\ (\mathit{prefixes}\ x)) \\
= \quad & \{\text{definition of } \mathit{we}\text{, filter, logic}\} \\
& \mathit{width}\ (\mathit{max}_{\preceq}\ (\mathit{we} \triangleleft (\mathit{prefixes}\ x)\ \cup\ \neg \mathit{we} \triangleleft (\mathit{prefixes}\ x)) \\
= \quad & \{\text{since } \mathit{we}\ x\text{, both lists are non-empty; (P.6) and max }\} \\
& \mathit{width}\ (\mathit{max}_{\preceq}\ (\mathit{we} \triangleleft (\mathit{prefixes}\ x))) \\
\geq \quad & \{\text{taking the maximum of a non-empty set of lists satisfying a predicate} \\
& \text{results in a list that still satisfies that predicate}\} \\
& L.
\end{aligned}$$

□

Proof of Lemma 3.2 (Monotonicity of *mdp*)
The two properties to prove are as follows:

$$x \sqsubseteq y \quad \Rightarrow \quad \mathit{mdp}\,x \sqsubseteq \mathit{mdp}\,y, \qquad \text{(P.14)}$$
$$x \sqsubseteq y \quad \Rightarrow \quad \mathit{mdp}\,x \preceq \mathit{mdp}\,y. \qquad \text{(P.15)}$$

If $x = y$ we are done. We thus assume $x \sqsubset y$, in which case we can decompose y into $y = x + a : z$ (for some possibly empty z).

For (P.14) we reason:

$$\mathit{mdp}\ x \sqsubseteq \mathit{mdp}\ (x + a : z)$$

$\Leftrightarrow \quad \{ \text{ (S.3) and (S.2) } \}$
$\quad mdp\ x \sqsubseteq mdp\ x\ \uparrow_{\preceq}\ max_{\preceq}\,(map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ z))$
$\Leftarrow \quad \{ \text{ see below } \}$
$\quad mdp\ x \sqsubseteq mdp\ x \quad \wedge \quad mdp\ x \sqsubseteq max_{\preceq}\,(map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ z)))$
$\Leftarrow \quad \{ \sqsubseteq \text{ reflexive } \}$
$\quad mdp\ x \sqsubseteq max_{\preceq}\,(map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ z)))$
$\Leftarrow \quad \{ \ mdp\ x \sqsubseteq x \ \}$
$\quad x \sqsubseteq max_{\preceq}\,(map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ z)))$
$\Leftarrow \quad \{ \text{ since } x \sqsubseteq x +\!\!+ w \text{ for all } w \ \}$
$\quad true.$

The second step is true simply because $a \uparrow_{\preceq} b$ is either a or b, and for any predicate P, $P\,(a \uparrow_{\preceq} b)$ is implied by $P\,a \wedge P\,b$.

For (P.15), we reason:

$\quad mdp\ x \preceq mdp\ (x +\!\!+ a : z)$
$\Leftrightarrow \quad \{ \text{ (S.3) and (S.2) } \}$
$\quad mdp\ x \preceq mdp\ x \uparrow_{\preceq} max_{\preceq}\,(map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ z))$
$\Leftarrow \quad \{ \text{ since } a \preceq a \uparrow_{\preceq} b \ \}$
$\quad true.$

Note that the first of these monotonicity properties (P.14) is not quite as trivial as it looks, as it depends on *mdp* consistently choosing the shortest, amongst several maximally dense prefixes.
□

Proof of Lemma 3.3
Recall that

- $mdp = max_{\preceq} \cdot prefixes$,
- the function *prefixes* returns shorter prefixes first, and
- $(\uparrow_{\preceq})$ is biased toward the left, thus, if $x_1 \uparrow_{\preceq} x_2 = x_2$, it must be the case that $x_1 \prec x_2$.

The list *mdp y* is a member of *prefixes y*. Since $(\uparrow_{\preceq})$ is left-biased, for any z appearing earlier than (or, to the left of) *mdp y* in *prefixes y*, we have $z \prec mdp\ y$. If $x \sqsubset mdp\ y$, x must appear earlier than *mdp y* in *prefixes y*. Thus we have $x \prec mdp\ y$.
□

Proof of Lemma 3.4 (Sandwich Lemma)
We need to prove that for any lists of elements x and y,

$$mdp\, y \sqsubseteq x \sqsubseteq y \quad \Rightarrow \quad mdp\, x = mdp\, y. \tag{P.16}$$

By (P.14) and $x \sqsubseteq y$ we have $mdp\, x \sqsubseteq mdp\, y$. We are thus left with proving $mdp\, y \sqsubseteq mdp\, x$. By applying (S.3) and then (S.2), we have

$$mdp\ y = mdp\ x\ \uparrow_{\preceq}\ max_{\preceq}\ (map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ y)).$$

Thus $mdp\ y$ is either $mdp\ x$ or $max_{\preceq}\ (map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ y))$. If it is the former case, we have $mdp\ y = mdp\ x \sqsubseteq mdp\ x$. The latter case, on the other hand, contradicts with the assumption that $mdp\, y \sqsubseteq x$, since no list in $map\ (x +\!\!+ [a] +\!\!+)\ (prefixes\ y)$ can be a prefix of x.
□

Proof of Lemma 3.6
The first requirement to prove is that for any list of elements x,

$$width\, x \leq L \quad \Rightarrow \quad x = wp\, x = mds\, x. \tag{P.20}$$

We will start with showing that $x = wp\ x$ by induction. The base case [ ] is trivial, following from the definition of wp. For the inductive case:

Case $a : x$

$$\begin{aligned}
& wp\ (a : x) \\
= \quad & \{\text{definition of } wp \text{ (P.17)}\} \\
& mdp\ (a : wp\ x) \\
= \quad & \{\text{induction, as } width\,(a : x) \leq L \Rightarrow\ width\, x \leq L, \text{ from (P.2) }\} \\
& mdp\ (a : x) \\
= \quad & \{\text{Lemma 3.1}\} \\
& a : x.
\end{aligned}$$

That $width\, x \leq L\ \Rightarrow\ x = mds\ x$ is also demonstrated using induction. The base case [ ] is trivial, following from the definition of mds. For the inductive case:

Case $a : x$

$$\begin{aligned}
& mds\ (a : x) \\
= \quad & \{\text{definition of } mds\} \\
& mdp\ (a : x)\ \uparrow_{\preceq} mds\ x \\
= \quad & \{\text{induction, as } width\,(a : x) \leq L \Rightarrow\ width\, x \leq L, \text{ from (P.2) }\} \\
& mdp\ (a : x)\ \uparrow_{\preceq}\ x
\end{aligned}$$

$$
\begin{array}{ll}
= & \{\text{Lemma 3.1}\} \\
& a : x \ \upharpoonright_{\preceq} \ x \\
= & \{\text{from the definition of } \preceq \text{ (see (P.6), (P.8)),} \\
& \text{and } \mathit{width}\,(a : x) \leq L \Rightarrow \ \mathit{width}\,x \leq L, \text{ from (P.2) } \} \\
& a : x.
\end{array}
$$

□

Proofs for Section 4

Proof of Lemma 4.2

We need to show that, given a list of elements x, if y is a densest prefix of x, then y is right-skew. We will take $0 < n < \mathit{length}\ y$ and demonstrate that $\mathit{take}\ n\ y \leq_d \mathit{drop}\ n\ y$. From the (S.$\leq_d$) density property, $\mathit{take}\ n\ y \leq_d \mathit{drop}\ n\ y$ is equivalent to $\mathit{take}\ n\ y \leq_d y$, which follows from $\mathit{take}\ n\ y \ \sqsubseteq \ x$, and y being a densest prefix of x. □

Proof of Lemma 4.3

We need to show that $z +\!\!+ z'$ is right-skew, if z and z' are both right-skew and $z \leq_d z'$. This requires that $\mathit{take}\ n\ (z +\!\!+ z') \leq_d \mathit{drop}\ n\ (z +\!\!+ z')$, for all n where $0 < n < \mathit{length}\ (z +\!\!+ z')$.

Case $n = \mathit{length}\ z$

Here, $\mathit{take}\,n\,(z +\!\!+ z') \leq_d \ \mathit{drop}\,n\,(z +\!\!+ z')$ simplifies to $z \leq_d z'$, which is true by assumption.

Case $n < \mathit{length}\ z$

$$
\begin{array}{ll}
& \mathit{take}\ n\ (z +\!\!+ z') \leq_d \mathit{drop}\ n\ (z +\!\!+ z') \\
\Leftrightarrow & \{\ \text{since } n < \mathit{length}\ z\ \ \} \\
& \mathit{take}\ n\ z \leq_d \mathit{drop}\ n\ z\ \ +\!\!+\ \ z' \\
\Leftrightarrow & \{\ \text{density property (S.}\leq_d\text{): (c)} \Leftrightarrow \text{(a), } \mathit{take}\ n\ z +\!\!+ \mathit{drop}\ n\ z = z\ \ \} \\
& \mathit{take}\ n\ z \leq_d z\ \ +\!\!+\ \ z' \\
\Leftarrow & \{\ \text{since } x \text{ is right-skew : } \mathit{take}\ n\ z \leq_d \mathit{drop}\ n\ z, \\
& \quad \text{and thus } \mathit{take}\ n\ z \leq_d z \text{ by Density Property (S.}\leq_d\text{): (c)} \Leftrightarrow \text{(a) } \} \\
& z \leq_d z\ \ +\!\!+\ \ z' \\
\Leftrightarrow & \{\ \text{density property (S.}\leq_d\text{): (a)} \Leftrightarrow \text{(c)}\ \ \} \\
& z \leq_d z' \\
\Leftrightarrow & \mathit{true}.
\end{array}
$$

Case $n > length\ z$
Symmetrical to the above case, using symmetrical density properties.
□

Proof of Lemma 4.4
We need to show that for z_1, z_2, z_3 non-empty list of elements such that $z_2 \leq_d z_3$,

$$z_1 \uparrow_d (z_1 +\!\!+ z_2) \uparrow_d (z_1 +\!\!+ z_2 +\!\!+ z_3) \;=\; z_1 \uparrow_d (z_1 +\!\!+ z_2 +\!\!+ z_3).$$

As the $\uparrow_d$ operator, in the case of a tie, chooses the left-hand argument, this lemma is proved as a corollary of the following lemma:

Lemma 4.4-a
Let z_1, z_2, z_3 be non-empty list of elements such that $z_2 \leq_d z_3$. Either $z_1 +\!\!+ z_2 \leq_d z_1$ or $z +\!\!+ z_2 <_d z +\!\!+ z_2 +\!\!+ z_3$.

Proof We split this into two cases, according to whether $z_1 \geq_d z_2$.
Case $z_1 \geq_d z_2$:
That $z_1 +\!\!+ z_2 \leq_d z_1$ follows from the density property (S.$\geq_d$): (c) $\Rightarrow$ (a).

Case $z_1 <_d z_2$: We reason:

$$\begin{array}{ll} & z_1 \leq_d z_2 \\ \Rightarrow & \{\ \text{Density property (S.}{<_d}\text{): (c)} \Rightarrow \text{(b)}\ \} \\ & z_1 +\!\!+ z_2 <_d z_2 \\ \Rightarrow & \{\ z_2 \leq_d z_3\ \} \\ & z_1 +\!\!+ z_2 <_d z_3 \\ \Rightarrow & \{\ \text{Density property (S.}{<_d}\text{): (c)} \Rightarrow \text{(a)}\ \} \\ & z_1 +\!\!+ z_2 <_d z_1 +\!\!+ z_2 +\!\!+ z_3. \end{array}$$

□

Proof of Theorem 4.5
We need to show that

$$max_{\uparrow_d}\ (map\ (z +\!\!+)\ (prefixes\ x)) \;=\; z \uparrow_d (z +\!\!+ x),$$

where z and x are non-empty lists of elements with x right-skew.

Intuitively, this is a straightforward generalisation of Lemma 4.4, where z corresponds to z_1 in that Lemma, and x to $z_2 +\!\!+ z_3$. A formal proof is more intricate, as a simple induction on x does not work, because x being right-skew doesn't mean that *tail* x is right-skew too.

Instead, we need to retain the information that the whole of x is right-skew, to provide sufficient context for the induction, which is what the following lemma does. Here, the variable y holds the "rest of x", so that the right-skew context can be provided. Theorem 4.5 then follows by letting $y = [\,]$.

Lemma 4.5-a
Let x, y and z be lists of elements such that z is non-empty and x,y satisfy:

$$(\forall n : (y +\!\!+ \mathit{take}\, n\, x) \neq [\,] \wedge n < \mathit{length}\, x \;:\; (y +\!\!+ \mathit{take}\, n\, x) \leq_d \mathit{drop}\, n\, x).$$

Then it is the case that

$$z \uparrow_d \mathit{max}_{\uparrow_d} (\mathit{map}\, (z +\!\!+ y +\!\!+)\, (\mathit{prefixes}\, x)) = z \uparrow_d (z +\!\!+ y +\!\!+ x).$$

Proof We prove the lemma by induction on x. The $[\,]$ case is trivially true. For the $a : x$ case, we reason:

$$\begin{array}{ll}
& z \uparrow_d \mathit{max}_{\uparrow_d} (\mathit{map}\, (z +\!\!+ y +\!\!+)\, (\mathit{prefixes}\, (a : x))) \\
= & \{ \text{ definition of } \mathit{prefixes}, \text{ definition of } \mathit{max}_{\uparrow_d} \; \} \\
& z \uparrow_d ((z +\!\!+ y) \uparrow_d \mathit{max}_{\uparrow_d} (\mathit{map}\, (z +\!\!+ y +\!\!+ [a] +\!\!+)\, (\mathit{prefixes}\, x))) \\
= & \{ \text{ by (S.1) } \} \\
& (z \uparrow_d (z +\!\!+ y)) \uparrow_d (z \uparrow_d \mathit{max}_{\uparrow_d} (\mathit{map}\, (z +\!\!+ y +\!\!+ [a] +\!\!+)\, (\mathit{prefixes}\, x))) \\
= & \{ \text{induction hypothesis, see below } \} \\
& (z \uparrow_d (z +\!\!+ y)) \uparrow_d (z \uparrow_d (z +\!\!+ y +\!\!+ [a] +\!\!+ x)) \\
= & \{ \text{ by (S.1), } \uparrow_d \text{ associative } \} \\
& z \;\uparrow_d\; z +\!\!+ y \;\uparrow_d\; z +\!\!+ y +\!\!+ [a] +\!\!+ x \\
= & \{ \text{ Lemma 4.4, since } y \leq_d (a : x) \; \} \\
& z \uparrow_d (z +\!\!+ y +\!\!+ [a] +\!\!+ x).
\end{array}$$

Above, we could perform induction because

$$\begin{array}{l}
(\forall n \,: (y +\!\!+ \mathit{take}\, n\, (a : x)) \;\neq [\,] \;\wedge\; n < \mathit{length}\, (a : x) \\
\quad : (y +\!\!+ \mathit{take}\, n\, (a : x)) \leq_d \mathit{drop}\, n\, (a : x)))
\end{array}$$

implies

$$\begin{array}{l}
(\forall n \,: (y +\!\!+ [a] +\!\!+ \mathit{take}\, n\, x) \;\neq [\,] \;\wedge\; n < \mathit{length}\, x \\
\quad : (y +\!\!+ [a] +\!\!+ \mathit{take}\, n\, x) \leq_d \mathit{drop}\, n\, x)).
\end{array}$$

□

Proof of Theorem 4.6
We need to show that

$$\begin{array}{l}
\mathit{max}_{\uparrow_d} (\mathit{map}\, (h +\!\!+)\, (\mathit{prefixes}\, (\mathit{concat}\, \mathit{xs}))) \quad = \\
\quad \mathit{max}_{\uparrow_d} (\mathit{map}\, ((h +\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, \mathit{xs})).
\end{array}$$

where all the segments of the partition xs are right-skew.

Proof This proceeds by performing induction on xs. The $[\,]$ case is trivial; for the inductive case $x : xs$, we reason:

$$
\begin{array}{ll}
& max_{\uparrow_d}\,(map\,(h{+\!\!+})\,(prefixes\,(concat\,(x : xs)))) \\
= & \quad \{\ \text{definition of } concat\ \} \\
& max_{\uparrow_d}\,(map\,(h{+\!\!+})\,(prefixes\,(x +\!\!+ concat\,xs))) \\
= & \quad \{\ \text{by (S.4)}\ \} \\
& max_{\uparrow_d}\,(map\,(h{+\!\!+})\,(prefixes\,x)) \uparrow_d \\
& \quad max_{\uparrow_d}\,(map\,((h +\!\!+ x{+\!\!+}))\,(prefixes\,(concat\,xs))) \\
= & \quad \{\ \text{Theorem 4.5}\ \} \\
& h \uparrow_d (h +\!\!+ x) \uparrow_d \\
& \quad max_{\uparrow_d}\,(map\,((h +\!\!+ x{+\!\!+}))\,(prefixes\,(concat\,xs))) \\
= & \quad \{\ \text{induction}\ \} \\
& h \uparrow_d (h +\!\!+ x) \uparrow_d \\
& \quad max_{\uparrow_d}\,(map\,((h +\!\!+ x{+\!\!+}) \cdot concat)\,(prefixes\,xs)) \\
= & \quad \{\ \text{definition of } concat\ \} \\
& h \uparrow_d (h +\!\!+ x) \uparrow_d \\
& \quad max_{\uparrow_d}\,(map\,((h{+\!\!+}) \cdot concat)\,(map\,(x :)\,(prefixes\,xs))) \\
= & \quad \{\ \text{definition of } prefixes,\ \uparrow_d \text{ idempotent and associative}\ \} \\
& max_{\uparrow_d}\,(map\,((h{+\!\!+}) \cdot concat)\,(prefixes\,(x : xs))).
\end{array}
$$

□

Proof of Theorem 4.9 (DRSP uniqueness)
This is by contradiction.
Let x be a non-empty list of elements, where $x = concat\ ps = concat\ qs$, and ps and qs are both different DRSPs of x, where

$$
\begin{array}{lll}
ps & = & [p_0, p_1, \ldots p_m] \\
qs & = & [q_0, q_1, \ldots q_n].
\end{array}
$$

Let i be the first index where ps and qs differ, i.e. $take\ i\ ps = take\ i\ qs$ but $take\ (i+1)\ ps \neq take\ (i+1)\ qs$. Thus we have $p_i \neq q_i$, and without loss of generality, let $p_i \sqsubset q_i$ (i.e. p_i is a proper prefix of q_i).

Consider where the end of q_i falls, amongst the segments $[p_{i+1}, \ldots p_m]$. Specifically, let p_j be the segment where q_i ends, such that there is a prefix $p'_j \sqsubseteq p_j$ so that $p_i +\!\!+ p_{i+1} +\!\!+ \ldots +\!\!+ p_{j-1} +\!\!+ p'_j = q_i$. Here $i < j$.

Either $p'_j \sqsubset p_j$ or $p'_j = p_j$; if the former, then from the right-skew property of p_j we have $p'_j \leq_d (p_j \backslash\backslash p'_j)$, which implies that $p'_j \leq_d p_j$ from the density property (S.$\leq_d$). Thus for either case, we have $p'_j \leq_d p_j$.

From ps being a DRSP of x, we have that $sdec\ (\leq_d)\ [p_i, p_{i+1} \ldots p_j]$ (recall that $sdec\ (\trianglelefteq)$ expresses a list being strictly decreasing with respect to the ordering $(\trianglelefteq)$). From above, it thus follows that $sdec\ (\leq_d)\ [p_i, p_{i+1} \ldots p_{j-1}, p'_j]$.

Note that for any lists of elements such that $a >_d b >_d c$, the density property (S.$>_d$) implies that $a \mathbin{+\!\!+} b >_d c$. Applying this step repeatedly to $p_i >_d p_{i+1} >_d \ldots >_d p_{j-1} >_d p'_j$ produces the result $p_i \mathbin{+\!\!+} p_{i+1} \mathbin{+\!\!+} \ldots \mathbin{+\!\!+} p_{j-1} >_d p'_j$. However, this contradicts the right-skew property of q_i.

Thus the DRSP of a sequence is unique. □

Proof of Theorem 4.10 (DRSP rotation)
We need to show that for any non-empty list of elements x,

$$\mathit{drsp}_{\leq_d}\ x \ = \ \mathit{reverse}\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x))).$$

From Theorem 4.9, it suffices to show that the partition on the right-hand side of the above equation is a DRSP with respect to the relation $\leq_d$, and then uniqueness ensures that this is the same as $\mathit{drsp}_{\leq_d}\ x$.

From the definition of a DRSP, we need to show that

(i) $\mathit{all}\ (\mathit{rightskew}\ (\leq_d))\ (\mathit{reverse}\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x))))$

(ii) $\mathit{sdec}\ (\leq_d)\ (\mathit{reverse}\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x))))$

For (i), we reason

$$\begin{array}{ll}
 & \mathit{all}\ (\mathit{rightskew}\ (\leq_d))\ (\mathit{reverse}\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x)))) \\
\Leftrightarrow & \{\ \text{property of } \mathit{all}\text{: } \mathit{all}\ p \cdot \mathit{reverse} \ = \ \mathit{all}\ p\ \} \\
 & \mathit{all}\ (\mathit{rightskew}\ (\leq_d))\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x))) \\
\Leftrightarrow & \{\ \text{distribution of } \mathit{all} \text{ over composition}\ \} \\
 & \mathit{all}\ (\mathit{rightskew}\ (\leq_d) \cdot \mathit{reverse})\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x)) \\
\Leftrightarrow & \{\ \text{property of } \mathit{rightskew}/\mathit{reverse}\text{; claim (see below)}\ \} \\
 & \mathit{all}\ (\mathit{rightskew}\ (\geq_d))\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x)) \\
\Leftrightarrow & \{\ \text{by definition, } \mathit{drsp}_{\geq_d}\ \mathit{xs} \text{ implies that } \mathit{all}\ (\mathit{rightskew}\ (\geq_d))\ \mathit{xs}\ \} \\
 & \mathit{true}.
\end{array}$$

For (ii), we reason

$$\begin{array}{ll}
 & \mathit{sdec}\ (\leq_d)\ (\mathit{reverse}\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x)))) \\
\Leftrightarrow & \{\ \text{property of } \mathit{sdec} \text{ and } \mathit{reverse}\ \} \\
 & \mathit{sdec}\ (\geq_d)\ (\mathit{map\ reverse}\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x))) \\
\Leftrightarrow & \{\ \text{claim, see below}\ \} \\
 & \mathit{sdec}\ (\geq_d)\ (\mathit{drsp}_{\geq_d}\ (\mathit{reverse}\ x)) \\
\Leftrightarrow & \{\ \text{by definition, } \mathit{drsp}_{\geq_d}\ \mathit{xs} \text{ implies that } \mathit{sdec}\ (\geq_d)\ \mathit{xs}\ \} \\
 & \mathit{true}.
\end{array}$$

Both the above claims require that the density comparison relation $\le_d$ has the property that reversing a list does not alter its density. That is, for any list y, it is the case that $y \le_d (reverse\ y) \le_d y$. This holds for both of the density comparison relations given for the MDS and MMS problems. Note however that this theorem is not required for the proof of our algorithm; it is included to illustrate a property of DRSPs. □

Proof of Lemma 4.11
Let y' be the longest right-skew prefix of y, a non-empty list of elements. (Such a prefix does exist, as any singleton list is right-skew.) We need to show that y' is the longest of the maximally dense prefix(es) of y, i.e. that $y' = max_{\upharpoonright_d}\ (prefixes\ y)$.

From Lemma 4.2 we have that every maximally dense prefix of y is right-skew. Therefore it suffices to show that y' is a maximally dense prefix of y. We proceed by seeking a contradiction: now assume that y' is not itself a densest prefix of y.

Let z be a densest prefix of y. As z is right-skew (Lemma 4.2), it must be a non-empty proper prefix of y', as y' is the longest right-skew prefix and $z \neq y$ (as z is a densest prefix of y, unlike z). Then

$$
\begin{aligned}
& \quad rightskew\ y' \\
\Rightarrow & \quad \{ \text{ as } z \sqsubset y', \text{ let } z +\!\!+\, z' = y'; \text{ definition of right-skew } \} \\
& \quad z \le_d z' \\
\Leftrightarrow & \quad \{ \text{ density property } (\mathrm{S}.\le_d) \ \} \\
& \quad z \le_d z +\!\!+\, z' = y'.
\end{aligned}
$$

That $z \le_d y'$ contradicts the assumption that y' is not a densest prefix of y, but z is a densest prefix of y. □

Proof of Theorem 4.12 (DRSP construction)
Let x be a list of elements. We will first show that $drsp_{\le_d}\ x$ can be constructed by repeatedly taking longest maximum-density prefixes. To be precise, let $ps = drsp1\ x$, where

$$
\begin{aligned}
& drsp1 \quad :: [Elem] \rightarrow [[Elem]] \\
& drsp1\ \ [\,] = [\,] \\
& drsp1\ \ x\ \ = y : drsp1\ (drop\ (length\ y)\ x) \\
& \qquad\qquad\quad \mathbf{where}\ y = max_{\upharpoonright_d}\ (prefixes\ x).
\end{aligned}
$$

We will show that ps satisfies the definition of a DRSP, by induction on the length of ps.

Case $ps = [\,] = x$
Trivial.

Case ps is a singleton, and thus $ps = [x]$.
From the definition of *drsp1*, this means that $x = max_{\upharpoonright_d}$ (*prefixes* x). Thus by Lemma 4.11, x is also right-skew. Thus $[x]$ is a DRSP.

Case Let $ps = (y : z : xs)$.
Here, $y = max_{\upharpoonright_d}$ (*prefixes* x) and $(z : xs) = drop\ (length\ y)\ x$, from the definition of *drsp1*. From the induction hypothesis, we can assume that $(z : xs)$ is a DRSP. All that remains to show, from the definition of DRSP, is that y is right-skew, and $y >_d z$.

As $y = max_{\upharpoonright_d}$ (*prefixes* x), by Lemma 4.11, y is also right-skew.

To show that $y >_d z$, consider $y ++ z$, which is a prefix of x. As y is the longest maximally-dense prefix of x, we have that $y >_d y ++ z$, which by (S.$>_d$) is equivalent to $y >_d z$.

We have proved part (i) of Theorem 4.12. Part (ii) follows, as taking longest maximally-dense prefixes is the same as taking longest right-skew prefixes, by Lemma 4.11.

Parts (iii) and (iv) follow by use of Theorem 4.10 (DRSP rotation): taking the longest minimally-dense suffix of x with respect to $\leq_d$ yields the reverse of taking the longest maximally-dense prefix of *reverse* x with respect to $\geq_d$. Also, if a list satisifies *rightskew* $(\leq_d)$, then its reverse satisfies *rightskew* $(\geq_d)$. □

Proof of Lemma 4.13
We have to show that *prepend* z ys is a DRSP, where z is a right-skew list of elements, ys is a DRSP, and

$$\begin{array}{lcl} prepend & :: & [Elem] \to [[Elem]] \to [[Elem]] \\ prepend\ z\ [\,] & = & [z] \\ prepend\ z\ (y : ys) & = & \mathbf{if}\ z \leq_d y\ \mathbf{then}\ prepend\ (z ++ y)\ ys \\ & & \mathbf{else}\ z : y : ys. \end{array}$$

This is a straightforward induction on ys.

Case $[\,]$
Trivial.

Case $ys = y : ys'$ and $z >_d y$
In this case *prepend* returns $z : y : ys'$, which satisfies the definition of a DRSP since $z >_d y$ and z is right-skew, and $y : ys'$ is a DRSP.

Case $ys = y : ys'$ and $z \leq_d y$.
By definition, ys' is still a DRSP. By Lemma 4.3 and $z \leq_d y$, we know that $z ++ y$ is right-skew. Therefore, by induction, *prepend* $(z ++ y)$ zs is a DRSP.
□

Proof of Lemma 4.14
We need to show that the following list has strictly increasing densities:

$$map\,((h{+\!\!+})\cdot concat)\,(prefixes\,(xs +\!\!+ [x]))$$

where h is a non-empty list of elements, $xs +\!\!+ [x]$ is a list of non-empty lists of elements having strictly decreasing densities, and $h +\!\!+ concat\,xs <_d x$.

Proof We use an induction on xs but decompose the list from the right.

Case $[\,]$
We have to prove that $h <_d h +\!\!+ x$. This follows from the given assumption that $h <_d x$, and the density property (S.$<_d$).

Case $xs +\!\!+ [y]$
The given assumptions are that $h +\!\!+ concat\,xs +\!\!+ y <_d x$, and that $xs +\!\!+ [y, x]$ has strictly decreasing densities. We expand the list in question:

$$\begin{aligned}
& map\,((h{+\!\!+})\cdot concat)\,(prefixes\,(xs +\!\!+ [y] +\!\!+ [x])) \\
= \; & map\,((h{+\!\!+})\cdot concat)\,(prefixes\,xs +\!\!+ [xs +\!\!+ [y], xs +\!\!+ [y, x]]) \\
= \; & map\,((h{+\!\!+})\cdot concat)\,(prefixes\,xs){+\!\!+} \\
& \qquad [(h +\!\!+ concat\,xs +\!\!+ y), (h +\!\!+ concat\,xs +\!\!+ y +\!\!+ x)].
\end{aligned}$$

To show that this list is strictly increasing, we have to show

(i) that $(h +\!\!+ concat\,xs +\!\!+ y) <_d (h +\!\!+ concat\,xs +\!\!+ y +\!\!+ x)$, and

(ii) that the list $map\,((h{+\!\!+})\cdot concat)\,(prefixes\,(xs +\!\!+ [y]))$ is strictly increasing.

Note that (i) follows immediately from the given assumption and the density property (S.$<_d$).

For (ii), this follows directly from induction, provided that we can show that $(h +\!\!+ concat\,xs) <_d y$:

$$\begin{aligned}
& (h +\!\!+ concat\,xs +\!\!+ y) <_d x \\
\Rightarrow \; & \quad \{\ xs +\!\!+ [y, x] \text{ has strictly decreasing densities (assumption), so } y >_d x\ \} \\
& (h +\!\!+ concat\,xs +\!\!+ y) <_d y \\
\Leftrightarrow \; & \quad \{\ \text{Density property S.}<_d\text{: (b)} \Rightarrow \text{(c)}\ \} \\
& (h +\!\!+ concat\,xs) <_d y.
\end{aligned}$$

□

Proof of Theorem 4.15
We need to show that

$$h +\!\!+ concat\ (maxchop\ h\ xs) \;=\; max_{1_d}\ (map\ ((h{+\!\!+})\cdot concat)\ (prefixes\ xs)),$$

where h is a non-empty list of elements, and xs a list of non-empty segments having strictly decreasing densities.

We use induction on xs by decomposing it from the right.

Case Empty list, $xs = [\,]$

This follows straightforwardly from the definitions.

Case Non-empty, of the form $xs +\!\!+ [x]$, and where $(h +\!\!+ \mathit{concat}\, xs) <_d x$

$$
\begin{array}{ll}
 & \mathit{max}_{\uparrow_d}\ (\mathit{map}\ ((h+\!\!+) \cdot \mathit{concat})\ (\mathit{prefixes}\ xs)) \\
= & \quad \{ \text{ Lemma 4.14, as } (h +\!\!+ \mathit{concat}\, xs) <_d x \ \} \\
 & h +\!\!+ \mathit{concat}\, xs +\!\!+ x \\
= & \quad \{ \text{ definition of } \mathit{maxchop}, \text{ when } (h +\!\!+ \mathit{concat}\, xs) <_d x \ \} \\
 & h +\!\!+ \mathit{concat}\, (\mathit{maxchop}\, h\, (xs +\!\!+ [x])).
\end{array}
$$

Case Non-empty, of the form $[x]$, and $h \geq_d x$

$$
\begin{array}{ll}
 & \mathit{max}_{\uparrow_d}\, (\mathit{map}\, ((h+\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, [x])) \\
= & \quad \{\ \mathit{prefixes}, \mathit{map}\ \} \\
 & \mathit{max}_{\uparrow_d}\, [h, h +\!\!+ x] \\
= & \quad \{ \text{ as } h \geq_d x \ \} \\
 & h \\
= & \quad \{ \text{ definition of } \mathit{maxchop} \ \} \\
 & h +\!\!+ \mathit{concat}\, (\mathit{maxchop}\, h\, [x]).
\end{array}
$$

Case At least two elements, of the form $xs +\!\!+ [y, x]$, and where $(h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \geq_d x$

We reason:

$$
\begin{array}{ll}
 & \mathit{max}_{\uparrow_d}\, (\mathit{map}\, ((h+\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, (xs +\!\!+ [y, x]))) \\
= & \quad \{ \text{ expanding } \mathit{prefixes}, \mathit{map}, \mathit{max}_{\uparrow_d}, \text{ etc, associativity of } \uparrow_d \ \} \\
 & \mathit{max}_{\uparrow_d}\, (\mathit{map}\, ((h+\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, xs))) \uparrow_d \\
 & \quad ((h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \uparrow_d (h +\!\!+ \mathit{concat}\, xs +\!\!+ y +\!\!+ x)) \\
= & \quad \{ \text{ from } (\text{S.}\geq_d), (h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \geq_d x \text{ is equivalent to } \} \\
 & \quad \{\ (h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \geq_d (h +\!\!+ \mathit{concat}\, xs +\!\!+ y +\!\!+ x) \ \} \\
 & \quad \{ \text{ and } \uparrow_d \text{ prefers left-hand argument in event of a tie } \} \\
 & \mathit{max}_{\uparrow_d}\, (\mathit{map}\, ((h+\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, xs)) \uparrow_d (h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \\
= & \quad \{ \text{ definitions of } \mathit{prefixes}, \mathit{map}, \mathit{max}, \text{ etc } \} \\
 & \mathit{max}_{\uparrow_d}\, (\mathit{map}\, ((h+\!\!+) \cdot \mathit{concat})\, (\mathit{prefixes}\, (xs +\!\!+ [y]))) \\
= & \quad \{ \text{ by induction } \} \\
 & h +\!\!+ \mathit{concat}\ (\mathit{maxchop}\, h\, (xs +\!\!+ [y]))
\end{array}
$$

$$
\begin{aligned}
= & \quad \{ \text{ definition of } \mathit{maxchop}, \text{ as } (h +\!\!+ \mathit{concat}\, xs +\!\!+ y) \geq_d x \ \} \\
& h +\!\!+ \mathit{concat}\, (\mathit{maxchop}\, h\, (xs +\!\!+ [y, x])).
\end{aligned}
$$

□

A *scanr*-Based Algorithm

While the final algorithm is given in the paper, we record here an alternative *scanr* version of the same algorithm.

In Section 3.6 we worked hard to prove that $mds = ms$, where ms is defined inductively. For the purpose of this section, it is more convenient to express ms as a $scanr$. It is not hard to show that:

$$ms = max_{\preceq} \cdot scanr\ (\lambda a\, x \rightarrow mdp\,(a : x))\ [\,]. \qquad \text{(S.5)}$$

Then, the derivation to introduce the same window structure goes as follows:

$$\begin{aligned}
& mds \\
= \quad & \{\ \text{proof of the outer algorithm, Section 3.6}\ \} \\
& ms \\
= \quad & \{\ \text{the above property of } ms \text{ (S.5)}\ \} \\
& max_{\preceq} \cdot scanr\ (\lambda a \rightarrow mdp \cdot (a :))\ [\,] \\
= \quad & \{\ \text{since } wflatten \cdot wbuild = id\ \} \\
& max_{\preceq} \cdot map\ wflatten \cdot map\ wbuild \cdot scanr\ (\lambda a \rightarrow mdp \cdot (a :))\ [\,] \\
= \quad & \{\ scanr\text{-fusion, see below}\ \} \\
& max_{\preceq} \cdot map\ wflatten \cdot scanr\ (\lambda a \rightarrow wmaxchop \cdot wcons\ a)\ (wbuild\,[\,]).
\end{aligned}$$

The *scanr*-fusion in the last step is merely a specialised version of *foldr*-fusion. A quick recap! Fusion for lists says that if you have

$$h\ (f\ a\ x) \quad = \quad g\ a\ (h\ x)$$

then you get that $h \cdot foldr\ f\ e\ = \ foldr\ g\ (h\ e)$. However, *scanr* is of course a particular instance of a *foldr*, and so if we specialise the above fusion to using a *scanr* and a *map*, we end up with the following: if we have that

$$k\ (f\ a\ b) \quad = \quad g\ a\ (k\ b)$$

then we get that $map\ k \cdot scanr\ f\ e\ = \ scanr\ g\ (k\ e)$.

Instantiating that for the step in the above derivation, we get that for any element a, we require that:

$$wbuild \cdot mdp \cdot (a :) = wmaxchop \cdot wcons\, a \cdot wbuild. \qquad \text{(P.26)}$$

This fusion condition (P.26) is established in the main paper.