Generalized Effect Algebras as Models of Concurrent Resources

Peter Jipsen    M. Andrew Moshier

School of Computational Sciences and Center of Excellence in Computation, Algebra and Topology (CECAT) Chapman University, Orange, California

BLAST2015@UNT
Outline

- Small partial groupoids
- Separation algebras
- Applications
A **groupoid** is a set $A$ with a binary operation $\cdot : A \times A \to A$

If $A$ is finite, say $A = \{a_1, \ldots, a_n\}$, then a groupoid can be defined by its operation table

<table>
<thead>
<tr>
<th></th>
<th>$a_1$</th>
<th>$a_2$</th>
<th>$\ldots$</th>
<th>$a_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
<tr>
<td>$a_n$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
</tbody>
</table>

← fill out with (some of) $a_1, \ldots, a_n$ any way you like
Introduction

- What do we know about 2-element groupoids?
- How many are there? 16 Up to isomorphism? 10

\[
\begin{array}{c|cc}
\cdot & 0 & 1 \\
\hline
0 & 0 & 0 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
C & 0 & 1 \\
\hline
0 & 0 & 0 \\
1 & 0 & 1 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\cdot & 0 & 1 \\
\hline
0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\pi_1 & 0 & 1 \\
\hline
0 & 0 & 0 \\
1 & 1 & 1 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\pi_2 & 0 & 1 \\
\hline
0 & 0 & 1 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\bar{\pi}_1 & 0 & 1 \\
\hline
0 & 1 & 1 \\
1 & 0 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\bar{\pi}_2 & 0 & 1 \\
\hline
0 & 1 & 0 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
xy = zw & \text{assoc.} \\
\text{comm.} & \text{reduct} \\
\text{idem.} & \text{of BA} \\
\end{array}
\]

\[
\begin{array}{c|cc}
\pi_2 & 0 & 1 \\
\hline
0 & 0 & 1 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\bar{\pi}_1 & 0 & 1 \\
\hline
0 & 1 & 1 \\
1 & 0 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
\bar{\pi}_2 & 0 & 1 \\
\hline
0 & 1 & 0 \\
1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c|cc}
xy = x & \text{implic.} \\
(Lz sg) & \text{reduct} \\
\text{of BA} & \\
\end{array}
\]

\[
\begin{array}{c|cc}
xy = y & \text{assoc.} \\
(Rz sg) & \text{BA} \\
\text{axioms} & \\
\end{array}
\]

\[
\begin{array}{c|cc}
xy = yy & \text{BA} \\
x^2x^2 = x & \\
\end{array}
\]

\[
\begin{array}{c|cc}
x^2x^2 = x & \text{BA} \\
\end{array}
\]

- Give axioms for the varieties they generate
Introduction

In fact, quite a bit is known about 2-element algebras

Post lattice (from Schölzel 2010)
A partial groupoid is a set $A$ with a partial binary operation $\cdot : A \times A \rightrightarrows A$.

If $A = \{a_1, \ldots, a_n\}$, then a partial groupoid can be defined by a partially filled out operation table:

\[
\begin{array}{c|cccc}
\cdot & a_1 & a_2 & \cdots & a_n \\
\hline
a_1 & & & & \\
a_2 & & & & \\
\vdots & & & & \\
a_n & & & & \\
\end{array}
\]
What do we know about 2-element partial groupoids?

- How many are there? \(81 - 16 = 65\)
- Up to isomorphism? \(45 - 10 = 35\)
Axioms for the ISP classes they generate?

- Which of these generate a **finitely axiomatizable** ISP class?
- **Natural duality theory** applies to partial algebras (Davey [2006])
- Which of these partial groupoids are **dualizable**?
- For this talk: want the operation to have an **identity element** 0
- Two **total** groupoids: 2-element semilattice and 2-element group

<table>
<thead>
<tr>
<th>\lor</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>+_2</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

- How many **partial groupoids with an identity element** are there?

<table>
<thead>
<tr>
<th>+_1</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

- \( P_1 = \begin{array}{c|cc} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 \\ \end{array} \) **Is ISP(\( P_1 \)) finitely axiomatizable?** Dualizable?
Subalgebras, products, homomorphisms

- **A partial groupoid** is a set \( P \) with a partial binary operation 
  \[ \ast : P \times P \rightharpoonup P \]
- Ljapin and Evseev [1997] call it a **pargoid** for short
- Every pargoid can be extended to a groupoid \( \tilde{P} \) by adding one element \( \infty \notin P \)
- \( x\tilde{\ast} y = x \ast y \) if \( x \ast y \) is defined, and \( x\tilde{\ast} y = \infty \) otherwise
- \( \text{dom}(\ast) = \{(x, y) \mid x \ast y \neq \infty\} \)
- \( Q \subseteq P \) is a (partial) **subalgebra** if \( Q \) is closed under existing products
- \( \prod_{i \in I} Q_i \) is the **product**, \( \ast \) defined pointwise, exits **iff** it exists in all coords
- Note that \( \tilde{P} \times \tilde{Q} \neq \tilde{P} \times Q \)
- \( h : P \rightarrow Q \) is a **homomorphism** if \( h(x \ast y) = h(x) \ast h(y) \) for all \( (x, y) \in \text{dom}(\ast) \)
- **HSP** is defined using these operations
Terms and formulas are defined as usual

A term $t(x_1,\ldots,x_n)$ is defined iff all subterms are defined

An identity $s(x_1,\ldots,x_n) = t(x_1,\ldots,x_n)$ holds in a partial algebra $P$ if for all $x_1,\ldots,x_n \in P$ either both sides are undefined, or they are defined and equal

A quasiidentity $s_1 = t_1 \& \cdots \& s_n = t_n \implies s = t$ holds in $P$ if for all assignments that make the premises defined and equal, $s, t$ are defined and equal
Why bother with partial operations?

- Boole originally considered \( \cup \) undefined for overlapping sets
- Products of partial algebras are cartesian (not true with \( \tilde{P} \))
- Natural duality now allows partial operations and relations on both sides
- The main reason: Computer Science
- Consider the memory of a computer: a list of cells with values in them
- \((m_0, m_1, \ldots, m_i, \ldots)\) or more generally:
- a function \( m : L \to V \) from a set \( L \) of locations to \( V \) of values
- As a program runs, it is allocated some of these cells
- The part of memory used is called a heap \( h \), where \( h : L \mapsto V \) is a partial function
- If several programs run concurrently, they use separate heaps
Separation Algebras
Separation algebras

- Calcagno, O’Hearn, Yang [2007] define a separation algebra to be a cancellative commutative partial monoid
- i.e., of the form \((A, +, 0)\) where \(+ : A \times A \to A\) is
  - \(x + y = y + x\) (commutative)
  - \((x + y) + z = x + (y + z)\) (associative)
  - \(x + 0 = x\) (0 is the identity)
  - \(x + z = y + z \implies x = y\) (cancellative)
- Typical model: \(P_{L,V} = \{\text{all heaps (= partial functions } L \to V)\}\) and
  \[h + k = \begin{cases} h \cup k & \text{if } \text{dom}(h) \cap \text{dom}(k) = \emptyset \\ \text{undef.} & \text{otherwise} \end{cases}\]
- E.g., if \(L = \{0, 1\}\) and \(V = \{0, 1\}\) we have \(A = \{uu, u0, u1, 0u, 1u, 00, 01, 10, 11\}\) where \(h = ab\) is the heap that satisfies \(h(0) = a, h(1) = b; h(x) = u\) means undefined
- Define heap algebras \(= S(\{P_{L,V} | L, V \text{ are sets}\})\). Note that \(0 = uu\)
Heap algebra examples

- $P_{2,2} =
\begin{array}{cccccccc}
  + & uu & u0 & u1 & 0u & 1u & 00 & 01 & 10 & 11 \\
  uu & uu & u0 & u1 & 0u & 1u & 00 & 01 & 10 & 11 \\
  u0 & u0 & 00 & 10 & & & & & & \\
  u1 & u1 & 01 & 11 & & & & & & \\
  0u & 0u & 00 & 01 & & & & & & \\
  1u & 1u & 10 & 11 & & & & & & \\
  00 & 00 & & & & & & & & \\
  01 & 01 & & & & & & & & \\
  10 & 10 & & & & & & & & \\
  11 & 11 & & & & & & & & \\
\end{array}$

- Define $x \leq y$ if $x + z = y$ for some $z$ (the natural order)

- Can you find another (smaller) example? Guess what! $P_1 = P_{1,1}$

- $P_1 = \begin{array}{c|cc}
  + & 0 & 1 \\
  0 & 0 & 1 \\
  1 & 1 & \\
\end{array} = \begin{array}{c}
  1 \\
\end{array}$ $P_{1,n} = \begin{array}{c}
  1 \\
  2 \\
  3 \\
  \cdots \\
  n \\
\end{array}$
Heap algebras = ISP($P_1$)

Products of $P_1$ are Boolean lattice reducts with $x + y = x \lor y$ if $x \land y = 0$

What do the (partial) subalgebras of products of $P_1$ look like?

**Theorem**

*The class of heap algebras is ISP($P_1$)*

**Proof.**

$P_{L,V} \cong (V \cup \{u\})^L$ where $u \notin V$

1. $P_{1,V}$ is a subalgebra of $(P_1)^V$
2. Observe that $P_{L,V} = (P_{1,V})^L$
What is known about separation algebras

Let \( SA = \) quasivariety of separation algebras (canc. comm. partial monoids)

\( SA \) is larger than \( ISP(P_1) \) since \( \mathbb{Z}_2 \in SA \)

\( P_1 \) is positive, i.e., satisfies \( x + y = 0 \implies x = 0 \), which fails in \( \mathbb{Z}_2 \)

Positive separation algebras are also known as generalized effect algebras

**Fact**

\( x \leq y \) is a partial order in positive separation algebras

**Effect algebras** come from quantum logic, *Foulis and Bennett* [1994]

Effect algebras are positive SAs that have unary \('\) such that \( x + x' = 0' \)
Orthogonal separation algebras

Addition in $P_1$ is **orthogonal**, i.e., $x + x = x + x \implies x = 0$

Unit interval with truncated $+$ is a **non-orthogonal** pos. SA: $\frac{1}{2} + \frac{1}{2} = 1$

**Lemma**: Orthogonal separation algebras are positive.
Proof: If $x + y = 0$ then $x + (x + y)$ is defined, so $(x + x) + y$ is defined, hence $x + x = x + x$ so we get $x = 0$

Orthogonal separation algebras are also known as **generalized orthoalgebras**

$P_1$ is **coherent**, i.e., if $x + y$, $x + z$ and $y + z$ are defined, so is $(x + y) + z$

Example of a **non-coherent** orthogonal separation algebra: Take an 8-element BA and remove the top element

Coherent orthoalgebras are f.o. equivalent to orthomodular posets
Concrete separation algebras

Let \( U \) be any set and define \(+\) on \( \mathcal{P}(U) \) by
\[
X + Y = \begin{cases} 
X \cup Y & \text{if } X \cap Y = \emptyset \\
\text{undefined} & \text{otherwise}
\end{cases}
\]

Then \( (\mathcal{P}(U), +, \emptyset) \) is a coherent orthogonal separation algebra \( \cong P_1|U| \)

A **concrete separation algebra** is any partial algebra embedded in this powerset algebra

Hence the class of **concrete separation algebras** is \( \text{ISP}(P_1) \)

But this is **smaller** than the class of coherent orthogonal separation algebras

Is it **finitely axiomatizable**?
New quasiidentities

**Lemma:** There are at least two more quasiequations that hold in $P_1$ and are not consequences of previous axioms:

1. $x \leq y + z \& y \leq x + z \implies x = y$

2. $w + x = y + z \& w + y = u \& x + z = v \implies x = y$

**Proof** that 1. holds in $P_1$: Suppose $x \leq y + z \& y \leq x + z$ but $x \neq y$.

By symmetry can assume $x = 0$, $y = 1$.

Then $x \leq y + z$ implies $z = 0$ (since $y + z$ must be defined).

But now $1 = y \leq x + z = 0 + 0 = 0$ is a contradiction.

1. fails in this coherent orthogonal SA:

\[
\begin{array}{cccc}
& u & z & v \\
& x & & \\
o & y & & \\
0 & & & \\
\end{array}
\]

where only $u + x, y + z, x + z, y + v$ are defined
Similarly 2. \( w + x = y + z \) & \( w + y = u \) & \( x + z = v \) \( \implies \) \( x = y \) holds in \( P_1 \): Suppose \( x = 0, y = 1 \).

\[ w + x = y + z \] implies \( z = 0 \) and \( w = 1 \).

But now \( w + y = 1 + 1 \) is undefined, contradicting \( w + y = u \).

Below is a coherent orthogonal SA that satisfies 1. but fails 2.
ISP($P_1$) is not closed under $H$ 

- $(P_{1,2})^2 \cong \{ u0, 0u, 1u, u1 \}$ 

- and has a homomorphic image $\notin ISP(P_1)$
Heap algebras satisfy no congruence equations

- Consider the heap algebra $P_{1,n} = \begin{array}{c} 1 \quad 2 \quad 3 \quad \cdots \quad n \\ \downarrow \downarrow \downarrow \downarrow \\ 0 \end{array}$

- Can identify any two maximal elements without collapsing any others
- Can identify any maximal element with 0 without collapsing any others
- Therefore $\text{Con}(P_{1,n}) = \text{Eq}(n) = \text{the lattice of equivalence relations on an } n \text{ element set}$
- Any lattice equation fails in $\text{Eq}(n)$ for some $n$
Applications
Applications of Separation Algebras

Let $L$ be a set of memory locations, and $V$ a set of values that they can store.

A state of a computation is a partial function $s : L \rightarrow V$

The set $S$ of all states is $(V \cup \{u\})^L$

A program $P$ is a binary relation on $S$, i.e., $P \subseteq S \times S$

The identity program $1 = \{(s,s) \mid s \in S\}; \quad \text{abort} = 0 = \{\}$

Operations on programs:

- Composition $PQ = P; Q = \{(r,t) \mid \exists s (r,s) \in P \text{ and } (s,t) \in Q\}$
- Nondeterministic choice $P + Q = P \cup Q$
- Finite iteration $P^* = 1 \cup P \cup PP \cup PPP \cup \ldots$
An abstract algebra of programs

A Kleene algebra is of the form \((A, +, 0, \cdot, 1, ^*)\) such that

- \((A, +, 0)\) is a join semilattice with bottom,
- \((A, \cdot, 1)\) is a monoid,
- \(x(y + z) = xy + xz, \quad (x + y)z = xz + yz,\)
- \(0x = x0 = 0, \quad 1 + x + x^* x^* = x^*,\)
- \(xy \leq y \implies x^* y \leq y \quad \text{and} \quad yx \leq y \implies yx^* \leq y\)

To handle if ... then ... else ... and while ... do Kozen [1997] defined

Kleene algebras with test \((A, A', +, 0, \cdot, 1, ^*, \overline{\cdot})\) where in addition \((A', +, 0, \cdot, 1, \overline{\cdot})\) is a Boolean algebra and \(A' \subseteq A\)

Now: if \(b\) then \(p\) else \(q = bp + \overline{b}q\) and while \(b\) do \(p = (bp)^* \overline{b}^\prime\)
A computation trace model

Let $S^* = \{s_1s_2\cdots s_n \mid s_i \in S \text{ for } i = 1,\ldots,n \in \mathbb{N}\}$, $1 = \{\lambda\}$ where $\lambda$ is the empty string, and for $u, v \in S^*$

define $u \diamond v \begin{cases} u_1 \cdots u_m v_2 \cdots v_n & \text{if } u_m = v_1 \\ \text{undefined} & \text{otherwise} \end{cases}$

So for example, $s \diamond s = s$, $sst \diamond tst = sstst$ and $sst \diamond sst = \text{undefined}$

For $P, Q \subseteq S^*$ let $PQ = \{u \diamond v \mid u \in P, v \in Q\}$, $1 = S$, $0 = \emptyset$,

$P + Q = P \cup Q$, $P^* = 1 \cup P \cup PP \cup PPP \cup \cdots$

For $B \subseteq S$, let $\overline{B} = S - B$

Then $(\mathcal{P}(S^*), \mathcal{P}(S), +, 0, \cdot, 1, ^*, \bigcirc)$ is a Kleene algebra with tests

Want to add concurrent composition to this model
Concurrent Kleene algebra with tests

Recall that the set of states $S$ contains all partial functions from $L$ to $V$.

So $(S, \oplus, 0)$ is the heap algebra $P_{L,V}$.

Let $(F_S, \cdot, ||, 1)$ be the free bi-monoid with generators $S$ and $x||y = y||x$.

Note: $\cdot$ and $|$ have the same identity $1$, but no other interaction.

$F\{\alpha, \beta\} = \{1, \alpha, \beta, \alpha\alpha, \alpha\beta, \beta\alpha, \beta\beta, \alpha||\alpha, \alpha||\beta, \beta||\beta, \alpha||\alpha\alpha, \alpha(\alpha||\alpha), \ldots\}$

Define the set of traces $G_S = S \cup \{\alpha u \beta \mid \alpha, \beta \in S, u \in F_S\}$.

and let $\alpha u \beta | \gamma v \delta = (\alpha \oplus \gamma)(u||v)(\beta \oplus \delta)$ if $\alpha \oplus \gamma, \beta \oplus \delta$ defined.

$\alpha | \gamma = \alpha \oplus \gamma$ and $\alpha | \gamma v \delta = (\alpha \oplus \gamma)v(\alpha \oplus \delta)$ (if defined).
Concurrent Kleene algebra with tests

Sequential composition of traces is as before:

\[ \alpha u \beta \diamond \gamma v \delta = \alpha u \beta \cdot v \delta \text{ if } \beta = \gamma, \text{ undefined otherwise} \]

Lift \( \diamond, | \) to sets of traces by \( P \cdot Q = \{ \alpha u \beta \diamond \gamma v \delta | \alpha u \beta \in P, \gamma v \delta \in Q \} \)

\( P | Q = \{ (\alpha u \beta | \gamma v \delta) | \alpha u \beta \in P, \gamma v \delta \in Q \}, \quad 1 = S, \quad \bar{B} = S - B \)

(\( \mathcal{P}(G_S), \mathcal{P}(S), \cup, \emptyset, \cdot, |, *, \cdot) \) is a concurrent Kleene algebra with test

More general concurrent KATs are defined with an arbitrary positive SA

A model of concurrency should also satisfy the weak exchange law

\[ (x|y)(z|w) \leq xz|yw, \text{ which can be done by ordering } F_S \text{ and } S \text{ and using only upsets of } G_S \text{ for the algebra} \]
Bunched implication algebras

A bunched implication algebra (BI-algebra) is of the form

\((A, \lor, \land, \rightarrow, \top, \bot, *, \backslash, /, 1)\)

where \((A, \lor, \land, \rightarrow, \top, \bot)\) is a Heyting algebra

(i.e. a bounded distributive lattice with \(x \land y \leq z\) iff \(y \leq x \rightarrow z\)) and

\((A, \lor, \land, *, \backslash, /, 1)\) is a commutative residuated lattice

(i.e. a commutative monoid with \(x * y \leq z\) iff \(y \leq x \backslash z\) iff \(x \leq z / y\))

If \((x \rightarrow \bot) \rightarrow \bot = x\) we get classical BI-algebras

CBI-algebras = commutative residuated Boolean monoids
= crm-algebras of Jónsson-Tsinakis [1993]

BI-algebras come from Separation Logic, a Hoare programming logic
for reasoning about pointers and concurrent resources
BI-algebras from separation algebras

Let \((P, \oplus, 0)\) be a positive separation algebra.

Recall the natural order \(x \leq y\) iff \(\exists z \; x \oplus z = y\).

\(Up(P)\) is the set of upward closed subsets of \(P\) = a completely distributive complete lattice under intersection and union.

Hence \((Up(P), \cup, \cap, \to, P, \emptyset)\) is a Heyting algebra.

Define \(X \ast Y = \{x \oplus y \mid x \in X, y \in Y\}\),

\(X \setminus Y = \{z \mid x \oplus z \in Y\) for all \(x \in X\}\), \(X / Y = Y \setminus X\) and \(1 = \{0\}\).

Then \((Up(P), \cup, \cap, \to, P, \emptyset, \ast, \setminus, /, 1)\) is a bunched implication algebra.

Let \(Up(SA) = \{Up(P) \mid P \in SA\}\).

**Problem:** Axiomatize the class \(HSP(Up(SA))\).
Natural duality (briefly)

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
</table>
| Is \( P_1 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 \end{pmatrix} \) dualizable in the sense of Davey [2006]?

A finite (partial) algebra \( A \) is **dualizable** if there exists a (topological) structure \( \bar{A} \) on the same set as \( A \) such that for all \( B \in ISP(A) \)

if we construct the dual \( \bar{B} = Hom(B, A) \) as a **closed substructure** of \( \bar{A}^B \)

then the **double dual** \( Hom(\bar{B}, \bar{A}) \) as a (partial) **subalgebra** of \( A^{\bar{B}} \) is **isomorphic** to \( B \)

The dual structure \( \bar{A} \) has **compatible fundamental (partial) function and relations**

E.g. \( \leq = \{00, 01, 11\} \) is compatible with \(+\) since \(00 + 01 = 01, \ 00 + 11 = 11\)
D-poset + ? as dual of $P_1$

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Let $D_1 = \begin{array}{ccc}0 & 0 & 0 \\ 1 & 1 & 0\end{array}$, the truncated minus operation on \{0, 1\}.

It is an example of a generalized D-poset

$- = \{000, 101, 110\}$ is compatible with $+$

$R = \{000, 001, 010, 011, 101, 011, 111\}$ is compatible with $+$

$Parity_n = \{v \in (P_1)^n | v \text{ has an even } \# \text{ of 1s}\}$

However, do not know if these (or all compatible) relations are sufficient for a duality.
References


D. Kozen, Kleene algebra with tests, Transactions on Programming Languages and Systems, 19, 3, 1997, 427–443

Thank you!

Peter Jipsen and Andrew Moshier — Chapman University — June 11, 2015