Group action morphisms in backtrack search

Christopher W. Monteith

October 22, 2008

1 Recapitulate

1.1 Combinatorial isomorphism problems

- What: Amounts to **deciding membership of orbits** under a group action.
- Our case: **Symmetric group** S_n.
- **Complicated** problems.
- **Practical solution** through partition backtrack algorithms.
- Vaguely: **determines a set** of permutations to search through.
- My study: nauty algorithm.
- Complicated methods for complicated problems: just look at the literature!
- McKay and Leon.

1.2 My goals

- Write a survey: literature highly specialised or too arcane.
- Reveal more of the forest: many treatments get bogged down in algorithmic details.
- May lead to connections being made between different methods.

1.3 What we want from the algorithm

- Group S_n and S_n-space Ω; left action.
- Canonical map:

$$\mu : \Omega \to \Omega$$

s.t.

$$(\forall \alpha \in \Omega)\ (\forall x \in S_n)\ \mu(\alpha) = \mu(x\alpha)$$

and

$$(\forall \alpha \in \Omega)\ (\exists y \in S_n)\ \mu(\alpha) = y\alpha.$$

- **Constant on each orbit** taking a value within the orbit.
- **No direct comparison** needed: $\alpha \simeq \beta \iff \mu(\alpha) = \mu(\beta)$.
- Furthermore, with perms x and y that give $x\alpha = \mu(\alpha) = y\beta$, we have $y^{-1}x\alpha = \beta$.
- **Automorphism group**:

$$\mathrm{Aut}\,(\alpha) := (S_n)_\alpha.$$

1.4 Ordered Partitions: Π_n

- Used to "encode" sets of partitions: easily retrievable.
- **Finite sequences**: disjoint, non-empty sets of integers; union to $[n]$.
- **Notn**: $(\pi_1, \pi_2, \ldots, \pi_k)$.
- **Acted** upon by S_n: **pointwise**.
- Each orbit contains a unique harmonious partition.
- **Harmonising** Function

$$h : \Pi_n \to \Pi_n$$

s.t.

$$(\forall \pi \in \Pi_n)\ (\forall x \in S_n)\ h(\pi) = h(x\pi)$$

and

$$(\forall \pi \in \Pi_n)\ (\exists y \in S_n)\ h(\pi) = y\pi$$

- **Canonical map**!
- **refinement relation** $\sqsubseteq$: $\pi \sqsubseteq \sigma \iff$ split cells in σ to obtain π.

2 Functions

2.1 Permutations defined by ordered partition

- Ordered partitions define a set of permutations.
- Set of all perms that harmonise a partition.
- **Grab them!** Define the function

$$\begin{aligned} \mathcal{B} : \Pi_n &\to \mathcal{P}(S_n) \\ \pi &\mapsto \{ x \in S_n \mid x\pi = h(\pi) \} \end{aligned}$$

- h canonical map $\Rightarrow \mathcal{B}(\pi)$ is always non-empty.
- **Prop:** $(\forall \pi \in \Pi_n)\, \mathcal{B}(\pi) = y\mathrm{Aut}(\pi)$, where $y\pi = h(\pi)$.
- **Prop:** $(\forall \pi \in \Pi_n)\ (\forall x \in S_n)\ \mathcal{B}(x\pi) = \mathcal{B}(\pi)\, x^{-1}$.
  - h constant on orbits: $h(x\pi) = h(\pi)$
  - Let $y \in \mathcal{B}(\pi)$ giving $y\pi = h(x\pi)$
  - $\therefore (yx^{-1})(x\pi) = h(x\pi)$
  - By prop: $\mathcal{B}(x\pi) = (yx^{-1})\mathrm{Aut}(x\pi)$
  - Standard: $\mathrm{Aut}(x\pi) = x\mathrm{Aut}(\pi)\, x^{-1}$
  - Finally:

$$\mathcal{B}(x\pi) = (yx^{-1})(x\mathrm{Aut}(\pi)\, x^{-1}) = y\mathrm{Aut}(\pi)\, x^{-1} = \mathcal{B}(\pi)\, x^{-1}.$$

- Only thing used is the **canonical property**!

2.2 Group action morphisms

- What is a group action morphism? Why do we want them?
- Let Ω and Γ be S_n-spaces
- **Defn:** S_n-morphism is function $\phi : \Omega \to \Gamma$ s.t.

$$(\forall \alpha \in \Omega)\ (\forall x \in S_n)\, \phi(x\alpha) = x\phi(\alpha).$$

- $\mathcal{B}$ an example.
- Why? Two fold. Suppose that

$$f : \Omega \to \mathcal{P}(S_n)$$

s.t.

$$(\forall x \in S_n)\ (\forall \alpha \in \Omega)\ f(x\alpha) = f(\alpha)x^{-1}$$

- **Prop:** $\forall \alpha \in \Omega$, $\forall x \in S_n$

$$f(\alpha)\alpha = f(x\alpha)(x\alpha).$$

- Set at least as large as the orbit.
- **Prop:** Let $\alpha \in \Omega$; fix $x \in f(\alpha)$.

$$x\mathrm{Aut}(\alpha) \subseteq f(\alpha).$$

- We can retrieve the elements of $\mathrm{Aut}(\alpha)$ from $f(\alpha)$.
- Nauty **constructs and searches** such a set. Hopefully a small one.

2.3 Split, Choose, Refine

- Some important S_n-morphisms.
- Split: take an ordered partition; **force** an element into a **singleton cell**.

$$\mathcal{S} : \Pi_n \times [n] \to \Pi_n$$

- **Important:** $\mathcal{S}(\alpha, \pi) \sqsubset \pi$.
- Choose: **Choose** a **non-singleton** cell from a coarse ptn.

$$\mathcal{C} : \Omega \times \Pi_n \backslash \Phi_n \to \mathcal{P}([n])$$

- Refine: Split **zero or more cells** in place.

$$\mathcal{R} : \Omega \times \Pi_n \to \Pi_n.$$

- **Important:** $\mathcal{R}(\alpha, \pi) \sqsubseteq \pi$.
- All domains and codomains form S_n-spaces.
- All three functions must be S_n-morphisms.

3 Tree function

- Function
$$\mathcal{T} : \Omega \times \Pi_n \to \mathcal{P}\left(\Omega \times \Pi_n\right)$$
s.t.
$$\mathcal{T}(\alpha, \pi) = \{(\alpha, \mathcal{R}(\alpha, \pi))\}$$
if $\mathcal{R}(\alpha, \pi)$ is fine; otherwise,
$$\mathcal{T}(\alpha, \pi) = \{(\alpha, \mathcal{R}(\alpha, \pi))\} \bigcup \left(\bigcup_{v \in \mathcal{C}(\alpha, \mathcal{R}(\alpha, \pi))} \mathcal{T}(\alpha, \mathcal{S}(\mathcal{R}(\alpha, \pi), v)) \right).$$

- **Prop:** $\mathcal{T}$ is an S_n-morphism:
$\forall(\alpha, \pi) \in \Omega \times \Pi_n$, $\forall x \in S_n$
$$x\mathcal{T}(\alpha, \pi) = \mathcal{T}(x\alpha, x\pi).$$

- **Prop:** At least one (β, σ) in $\mathcal{T}(\alpha, \pi)$ such that σ is fine.

4 Harvesting what we need

4.1 Grabbing Permutations

- Turn the tree set into a set of permutations.

- Define
$$\begin{aligned} \mathcal{L} : \mathcal{P}\left(\Omega \times \Pi_n\right) &\to \mathcal{P}\left(S_n\right) \\ A &\mapsto \{\, x \in S_n \mid (\exists (\alpha, \pi) \in S)\ \mathcal{B}\left(\pi\right) = \{x\} \,\} \end{aligned}$$

- **Prop:** $\mathcal{L}$ is an S_n-morphism:
$\forall A \in \mathcal{P}\left(\Omega \times \Pi_n\right)$, $\forall x \in S_n$
$$\mathcal{L}\left(xA\right) = \mathcal{L}\left(A\right) x^{-1}.$$

- **Cor:** $\mathcal{L} \circ \mathcal{T}$ is an S_n-morphism:
  $\forall (\alpha, \pi) \in \Omega \times \Pi_n$, $\forall x \in S_n$

  $$\mathcal{L} \circ \mathcal{T}\,(x\alpha, x\pi) = \mathcal{L} \circ \mathcal{T}\,(\alpha, \pi)\, x^{-1}.$$

  (Morphisms compose to morphisms)

- $\mathcal{L} \circ \mathcal{T}$ is just an instance of f earlier.

4.2 Canonical Map

- For our canonical map:
  $\forall (\alpha, \pi) \in \Omega \times \Pi_n$, $\forall x \in S_n$

  $$\mathcal{L} \circ \mathcal{T}(\alpha, \pi)(\alpha, \pi) = \mathcal{L} \circ \mathcal{T}(x\alpha, x\pi)(x\alpha, x\pi).$$

- So select a unique representative from the set

  $$\mathcal{L} \circ \mathcal{T}(\alpha, \pi)(\alpha, \pi)$$

  (constant across entire orbit).

4.3 Automorphism Group

For our automorphism group: Fix x in $\mathcal{L} \circ \mathcal{T}(\alpha, \pi)$; we have

$$x\mathrm{Aut}\left((\alpha, \pi)\right) \subseteq \mathcal{L} \circ \mathcal{T}(\alpha, \pi).$$

4.4 Let's be realistic

- Need the set $\mathcal{L} \circ \mathcal{T}(\alpha, \pi)$ as small as we can get it.
- $\mathcal{R}$ is the key to better performance.
- Other "pruning" is possible when we get into algorithmic mode: uses auts that have already been found.
- Base and strong generating set methods need to be introduced.

5 Where to?

There are some opportunities to generalise:

- What if we are not acting on Ω with S_n?
- Group action morphisms are sufficient – but are they necessary?
- More on these next week.