SE36010 Engineering Knowledge Based Systems

Today: Agent Interaction
Ant Colony Optimisation

Thomas Jansen

March 6$^{\text{th}}$

Plans for Today

What happened so far. . .

Remember

- intelligent agents, for us rational agent
  rational $\widehat{=}$ maximising expected performance
- different agent programs to implement intelligent agents
  - table-driven
  - simple reflex
  - model-based reflex
  - goal-based
  - utility-based
  - learning

Observations

- sometimes more than just one agent involved
- scenarios with multiple agents may require agent communication to facilitate agent interaction

Departure from Single-Agent Environments

- multieffector planning
  single agent with multiple effectors with concurrent operation
  e. g., mobile robot with arms
- multibody planning
  single control unit ($\widehat{=}$ single agent) for multiple physically separated effectors ($\widehat{=}$ 'bodies')
  e. g., factory with multiple centrally controlled robots
- agents in competitive environments ($\widehat{=}$ agents' goals in conflict)
  e. g., games like chess
- cooperating agents ($\widehat{=}$ agents share one common goal)
  e. g., **one** team in robot football

Observation cooperation of agents requires some form of coordination
usually implying some form of information exchange
$\rightsquigarrow$ communication

Agent Interaction

Topics in multiagent scenarios

- design of rational agent
  $\rightsquigarrow$ game theory
- design of environments that maximise common good
  provided that agents act rationally
  $\rightsquigarrow$ mechanism design
- design of logical agents
  see next Tuesday
- planning
  see next Tuesday
- learning
  see next Wednesday
- communication
  rough overview and one example now

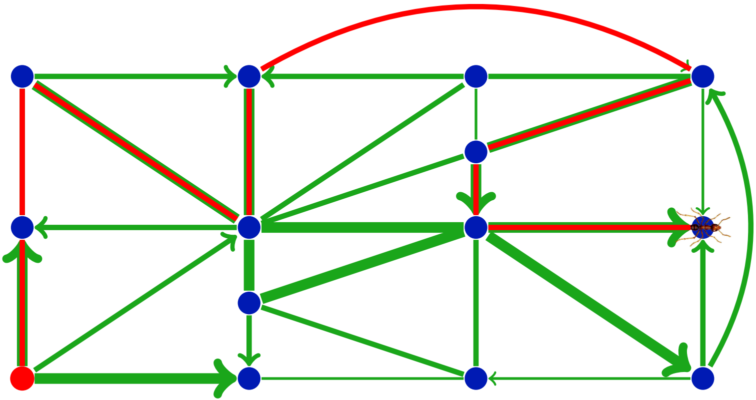
Agent Communication

- direct communication: messages
  - common language required
    (e. g., KIF (knowledge interchange format), CLIF (common logic interchange format), KQML (knowledge query and manipulation language), FIPA ACL (foundation for intelligent physical agents agent communication language))
  - coordination about times of communication required
- indirect communication: communication through interpreter
  - common language with interpreter needed
  - coordination about times of communication with interpreter required
- indirect communication: blackboard systems
  - common language needed
  - flexible with respect to times of communication
- indirect communication: stigmergy (changes in environment)
  example now

Ants

Bionics: From Ants to Algorithms

Consider directed graph with edge weights and a start node;
ant, performing a random walk, biased by edge weights,
starting in start, not repeating edges, ending in cul-de-sac



Random Walks of Ants on Weighted Graphs

Observations

- leaving pheromone equally on used edges reinforces ‘heavy edges’ ($\widehat{=}$ stabilising) and short paths
- classical algorithms find shortest paths much faster and more reliable $\rightsquigarrow$ Do not use this to compute shortest paths!
- paths may encode something entirely different
- if ‘something entirely different’ has a quality pheromone deposits can depend on this quality
- initial pheromone distribution ($\widehat{=}$ initial weights) is important

Initial pheromone distribution

- equal for each edge $\rightsquigarrow$ pure random walk on graph
- alternatively, heuristic based on ‘edge quality’, larger weights for ‘better edges’

A Bit More Algorithmic...

Basic Ant System

1. Initialise empty best-so-far solution x^*.
2. Initialise all pheromone values $\tau(e)$.
3. Repeat
4.   Construct one path x.
5.   Compute quality $f(x)$.
6.   If $f(x) > f(x^*)$ set $x^* := x$.
7.   Update pheromone values with respect to x^*.
8. Until stopping criterion tells you to stop.
9. Output x^*, $f(x^*)$.

Some conventions

- pheromone values are non-negative
- unless something known, all pheromone values equal initially

Construct One Path

Remember We have

- weighted directed graph $G = (V, E)$ with edge weights $\tau(e)$
- ants don't use edges twice

Definition for $v, w \in E$ we call (v, w) feasible
if $(v, w) \in E$ and $(v, w$ has not been used before

Construct one path

1. Initialise all $e \in E$ as unused.
2. Set $v :=$ start node.
3. Repeat
4. $\quad s := \sum_{w \in V, (v,w) \text{ feasible}} \tau(v, w)$
5. $\quad$ If $s > 0$ then
6. $\quad\quad$ Select (v, w) randomly with probability $\tau(v, w)/s$.
7. $\quad\quad$ Mark (v, w) as used. Set $v := w$.
8. Until $s = 0$

Update Pheromone Values

Remember We have

- weighted directed graph $G = (V, E)$ with edge weights $\tau(e)$
- one path $x = (e_1, e_2, \dots, e_l)$

Definition pheromone evaporation by factor $1 - \rho$ on all edges
pheromone increase on used edges by summand ρ
$(\rho \in (0, 1))$

Update pheromone values

1. For all $e \in E$ set $\tau(e) := (1 - \rho)\tau(e)$.
2. For all $e \in x$ set $\tau(e) := \tau(e) + \rho$

Observation pheromone update
independent from solution quality $f(x)$

$\rightsquigarrow$ influence somewhere else needed

A Very Simple Ant System

Basic Ant System

1. Initialise empty best-so-far solution x^*.
2. Initialise all pheromone values $\tau(e) := 1$.
3. Repeat
4. Construct one path x.
5. Compute quality $f(x)$.
6. If $f(x) \geq f(x^*)$ set $x^* := x$.
7. Update pheromone values with respect to x^*.
8. Until stopping criterion tells you to stop.
9. Output x^*, $f(x^*)$.

Observations

- pheromone update only if new best solution found
- '$f(x) \geq f(x^*)$' is very different from '$f(x) > f(x^*)$'
- algorithmically really simple

Update Pheromone Values Reconsidered

Update pheromone values

1. For all $e \in E$ set $\tau(e) := (1-\rho)\tau(e)$.
2. For all $e \in x$ set $\tau(e) := \tau(e) + \rho$

Observation pheromone values $\tau(e)$ can become
arbitrarily large $\rightsquigarrow$ 'always' selected
arbitrarily close to 0 $\rightsquigarrow$ 'never' selected
both undesriable

Consequence avoid this by introducing boundaries $\tau_{\min}$, $\tau_{\max}$
and have $\forall e \in E\colon \tau_{\min} \le \tau(e) \le \tau_{\max}$ all the time

A Simple 'Real' ACO: MMAS

Max-Min Ant System (MMAS)

1. Initialise empty best-so-far solution x^*.
2. Initialise all pheromone values $\tau(e) := \tau_{\min} + (\tau_{\max} - \tau_{\min})/2$.
3. Repeat
4. Construct one path x.
5. Compute quality $f(x)$.
6. If $f(x) \geq f(x^*)$ then
7. Set $x^* := x$.
8. For all $e \notin x^*$ set $\tau(e) := \max\{\tau_{\min}, (1-\rho)\tau(e)\}$.
9. For all $e \in x^*$ set $\tau(e) := \max\{\tau_{\min}, \min\{\tau_{\max}, (1-\rho)\tau(e) + \rho\}\}$.
10. Until stopping criterion tells you to stop.
11. Output x^*, $f(x^*)$.

Applying Ant Colony Optimisation

Remember ant colony optimisation

- has natural application in finding shortest paths
- but should not be used for computing shortest paths

How can we apply ant colony optimisation?

One option consider similar problems

- traveling salesperson problem (TSP)
- multi-objective shortest paths
- minimum spanning trees (MST)
- degree-restricted minimum spanning trees

Other option transform problems into graph-problems

Transforming Problems

Idea have problem given by set of instances I,
set of potential solutions S
function to measure quality of potential solutions $q\colon S \to \mathbb{R}$;
transform this into a graph G for ACO such that
a path x in G can be easily transformed into some $s \in S$

Observation would be nice to have a general scheme
instead of starting from scratch each time

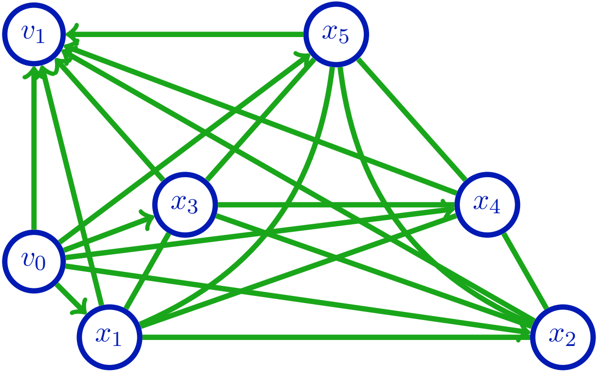
Observation canonical form for problems would be useful

Observation many problems can be described
as pseudo-Boolean functions $f\colon \{0,1\}^n \to \mathbb{R}$

Construction Graphs for Pseudo-Boolean Functions

Disc Graph

Example for $n = 5$



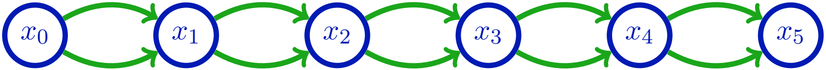
Path $\rightarrow x \in \{0, 1\}^n$

- paths start in v_0, end in v_1
- $x[i] = 1 \Leftrightarrow x_i$ on path
- allow edges to be used multiple times

Construction Graphs for Pseudo-Boolean Functions

Chain Graph

Example for $n = 5$

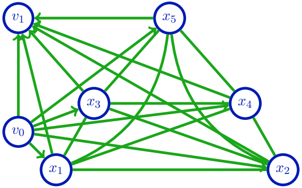


Path $\rightarrow x \in \{0, 1\}^n$

- paths start in x_0, end in x_n
- $x[i] = 1 \Leftrightarrow$ upper edge (x_{i-1}, x_i) on path

Comparing Construction Graphs

disc graph



chain graph

Which construction graph is better?

Observe probabilities $p_{G,x} = \mathrm{Prob}\,(\text{initially, construct } x \text{ in } G)$

- $p_{\text{chain},0^n} = \left(\frac{1}{2}\right)^n$, $p_{\text{chain},10^{n-1}} = \left(\frac{1}{2}\right)^n$, $p_{\text{chain},110^{n-2}} = \left(\frac{1}{2}\right)^n$
- $p_{\text{disc},0^n} = \frac{1}{n+1}$
- $p_{\text{disc},10^{n-1}} = \frac{1}{n+1} \cdot \frac{1}{n} = \frac{1}{n^2+n} = \Theta\left(\frac{1}{n^2}\right)$
- $p_{\text{disc},110^{n-2}} = \frac{1}{n+1} \cdot \frac{1}{n} \cdot \sum_{i=0}^{\infty} \left(\frac{1}{n}\right)^i \frac{1}{n} = \frac{1}{n^2+n} \cdot \frac{1}{n-1} = \Theta\left(\frac{1}{n^3}\right)$

Observation chain graph is less biased

Playing with MMAS

Consider a simple test problem 'find a target'
Pick target $t \in \{0,1\}^n$.
$f_t(x) := |\{i \in \{1,2,\dots,n\} \mid x[i] = t[i]\}|$

Test MMAS with $\tau_{\min} = 1/n$, $\tau_{\max} = 1 - 1/n$
on f_t (for arbitrary t) with chain graph as construction graph
1. for $n = 320$ with $\rho = 2^i/n$ for all $i \in \{0,1,2,\dots,\lfloor \log_2 n \rfloor\}$
2. for $\rho = 1/3$ with $n = 10 \cdot 2^i$ for all $i \in \{0,1,2,\dots,7\}$

doing 30 runs for each setting

Visualise results by plotting average number of times the main loop was executed until the target was found the first time
possibly displaying more info. about distribution of results
($\leadsto$ box-and-whisker plots, e. g., R (`www.r-project.org`))

Sources

Sources used today
in case you want to read more ore more directly

- S. Russel, P. Norvig (2010): *Artificial Intelligence. A Modern Approach.* 3rd edition. Section 11.4.
- D. Sudholt (2012): Theory of swarm intelligence. Tutorial. GECCO Companion. http://dx.doi.org/10.1145/2330784.2330938
- W. Guttjahr (2004): Chains, disks and drums: Ant colony optimization on diverse construction graphs. Extended Abstract. Dagstuhl Seminar 'Theory of Evolutionary Algorithms'. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.4.9019

Summary & Take Home Message

Things to remember

- multi-agent systems
- different forms of agent communication: direct (messages) or undirect (interpreter; blackboard; stigmergy)
- ant colony optimisation
- construction graphs

Take Home Message

- Multi-agent systems come with a number of challenging problems.
- Ant colony optimisation is a nature-inspired randomised search heuristic.
- Using a construction graph any optimisation problem can be turned into a problem that allows for the application of ant colony optimisation but not all should.