iciic script

[1]
I'd like to tallk about
"Solving Constraint Satisfaction Problems by Cunning Ants with Multi-Pheromones."
I'm Takuya Masukane from Takushoku University.

[2]
First of all, let me give the summary of this work.
Our purpose is to efficiently solve hard constraint satisfaction problems.
For the purpose,
we employ an algorithm based on ant colony optimization, ACO, with another pheromone.
As a result of this work,
our method can be effective to solve instances around the phase transition region.

[3]
In our work, the target is Graph Coloring Problem with 3 colors, 3COL.
It is a standard benchmark of the CSPs.
3COL is the problem to assign 3 colors to each vertex without violations.
In 3COL, the violation is defined as the same color at both ends of an edge, like here.
If coloring with no violations is found, the instance is solved.
Now, in this study, we define "constraint density," d.
It is the number of edges divided by the number of vertices.
For the prior work, in 3COL,
it is reported that hard instances concentrate, and it takes a lot of search cost to solve them
where d equals from 2.3 to 2.4.
So many methods have studied.

[4]
ACO is the one of the meta-heuristics to solve hard problem.
It is the imitation of ant's communication with their pheromone.
Our proposed model is based on ACO.
(1)
In real world, ants drop pheromone on the path from the feed to their nest.
To follow the pheromone, other ants return to their nest quickly.
This is "pheromone communication."
Then, the pheromone on the long path evaporates soon.
Conversely, the pheromone on the short path does not evaporate soon.
This process seems to be optimization.

[5]
Cunning Ant system is one of the algorithms based on ACO.
(1)
Each ant constructs assignment candidate probabilistically
according to the amount of pheromone, tau,
and the number of constraint violations, ita.
(2)
Then, pheromone is updated,
according to the evaporation rate, rho, and the best assignment.
Almost ACO algorithms assign to all vertices at each cycle.
So, construction of candidate has many time.
Then, cAS uses partial assignment at previous cycle.
And, it can reduce the search time.

[6]
In the process of search, depending on the pheromone state, cAS can tend to get caught in local optima.
It may be because assignment construction of cAS has only one aim.
It is to converge the assignment candidates to the assignment having less violations.
Then, in this study, in addition to same purpose as cAS, we set other aims.

[7]
Now, let me talk about our proposed model.
To achieve our purpose, in our model, we provide another pheromone.
We call it "negative pheromone."
To follow this pheromone, we should avoid an assignment which has many violations.
This pheromone is updated according to the worst assignment in the set.

[8]
And, this is our proposed algorithm, cAS with Negative pheromone.
We call it cASNEP.
The probability to assign value is different.
And we use two types of pheromones.
The negative pheromone is stored apart from usual pheromone.
Negative pheromone is updated according to the worst assignment.

[9]
To evaluate the effectiveness of our model, we make experiments.
We compare cAS and cASNEP.
Then, these are the results.
The red is cAS, and the blue is cASNEP.
First, about the percentage of solved instances,
we perform in several constraint density patterns.
In all cases, our proposed model overcome the naive cAS.
In particular, in phase transition region, rho equals 2.3 and 2.4,
the effectiveness is remarkable.

[10]
Then, next result is about search cost.
We perform the experiment in d equals 2.3.
The graph shows the cost for each trial which can solve.
We sort the trial in ascending order.
(1)
From this graph, almost solution is found by 25,000 constructions.
(2)
And in cASNEP, the number of solution found by 25,000 constructions is more than cAS.
Therefore, cASNEP should find solutions stably.

[11]
These results are difference in the situation.
Left figure shows the percentage of solved instances against the variation of rho.
In almost cases, the proposed model is superior.
Right figure shows the percentage of solved instances for larger instances.
The cAS has less progress.
However, the cASNEP grows gradually.

[12]
Finally, this is the conclusion.
We proposed a new model based on ACO to solve CSPs.
We provided another pheromone "negative pheromone."
Then, our proposed method is superior from the several points of view than naive method.
Our future work is to compare with other meta-heuristics,
and to apply to other types of CSPs.
Thank you.