PDF Content Masking Attack Against Information Based Online Serv ices

05/12/2020 19:48:13

Authors

Ian Markwood and Dakun Shen and Yao Liu and Zhuo Lu from University of South Florida

We present a new class of content masking attacks against the Adobe PDF standard causing documents to appear to humans dissimila r to the underlying content extracted by information based services We show three attack variants with notable impact on realworld s ystems Our first attack allows academic paper writers and reviewers to collude via subverting the automatic reviewer assignment syst ems in current use by academic conferences including INFOCOM which we reproduced Our second attack renders ineffective plagiar ism detection software particularly Turnitin targeting specific small plagiarism similarity scores to appear natural and evade detection In our final attack we place masked content into the indexes for Bing Yahoo and DuckDuckGo which renders as information entirely different from the keywords used to locate it enabling spam profane or possibly illegal content to go unnoticed by these search engin es but still returned in unrelated search results Lastly as these systems eschew optical character recognition OCR for its overhead we offer a comprehensive and lightweight alternative mitigation method

paper review

Definitely speaking, this paper including 6 parts.

- TFMP formulation
- comlexity of TFMP
- model variations
- insights from stucture
- Insights from computations
- conclusion

As the requirements asked. I will discuss the issues as below:

Firstly,

Motivation

In my opinion, when we do research or reading a paper. we should know:

What's the problem? Why you do it? what about the others have done? Then how you do it? How could you evaluate it? (such as make a comparation with others or yourself experiments.)

Problem in this paper.

Due to the demand for airport use has been increasing rapidly, however, airport capacity has been stagnating. Congestion on the air traffic system.

Contributions:

- 1、model that taking into account the (compara to others. or what about the others have done?)
  - capacities of NAS
  - capacities of airports
- 2、complexity of the problem is **NP-Hard** (point out its hard)
- 3、account for several variations of the basic problem
  how to reroute flights
  how to handle banks in the hub

- 4、compared to all others proposed in the literature(1994 a) for this problem
- 5、solve large scale, realistic size problems (several thousand flights).

problem definition

schedule flights in real time in order to minimize congestion costs

model formulation

- <u>**objective:**</u> when each flight held on the ground and in the air in order to minimize the total delay cost.
- <u>**input data:**</u>
  flights
  airports
  time periods
  continued flights
  path
  capacity(air sector or depart airport and arrival airport)
  scheduled time(depart and arrival)
  cost(ground and in the air)
  must time in one sector
- <u>**decision variables**</u>:
  flight f arrives at sector j by time t.
- <u>**the objective function:**</u> （<u>**formula**</u>） (<u>minus</u>)
  - actual departure time - scheduled departure time
    <u>**means**</u>
    <u>held on the ground time</u> (1)
- actual arrival time - scheduled arrival time -(actual departure time - scheduled departure time)
  <u>**equals**</u>
  (actual arrival time-actual departure time)-(scheduled arrival time-scheduled departure time)
  <u>**means**</u>
  <u>held in the air time</u> (2)

so the minimize total delay cost <u>**equals**</u>

(1) * cost on the ground +(2) * cost in the air (<u>mutiply</u> by)

solution algorithm

complexity of the tfmp is NP-hard problem

the author using a job-shop schedule to confirm all the constraints of the TFMP will be satisfied if and only if there is a feasible job-shop schedule.

modeling variations(could be used in many directions)

- Dependence Between Arrival and Departure Capacities Hub Connectivity with Multiple Connections
- Banks of Flights
- Rerouting of Aircraft

implementation

using cplex

case studies

Air Traffic Flow Management Problem Test Cases

performed one set of experiments in the <u>**four airports**</u> for 200 flights over a 24-hour time period and another set for 1,000 flights over a 24-hour time period.

- Boston Logan (BOS),
- NY LaGuardia (LGA),
- Washington National (DCA),
- a node representing all other airports(X)

For the set of 200 flights, to solve the problem CPLEX requires 234 seconds CPU time.

For the 1000, solve it at the infeasibility border over a 24 hour time period.

paper replication

algorithm design

I think it is a linear programming (LP).

main.cpp+tfmp.cpp+createInfo.cpp+commonFunction.cpp+ .h

Firstly, structure input

```
// create capacity Info.
        Create_capacity( D_k, A_k, S_sector, H, j);
 //create time Info. still have some problems
        Create_time( D_f, R_f, S_f, flight);
//create path
        Create_path( P, P_f, X, flight,j,env,H);
 //create the cost_g,cost_a,P_f
        Create_Cost( cost_g, cost_a, P.copy(), P_f.copy(), flight, t, env, X);
//create each flight's every sector's least-time
        Create_sector_time( L_sector, L_sector_f, X, flight,j,env,H,D_f.copy(),R_f.copy());
// create method
        create_sector_FeasibleTime(T_sector_first, T_sector_last, T_sector_first_f, T_sector_last_f, env, L_sector_f
```

Secondly,structure Objective function

Finally,structure Limit function (7 formula)

data structure

```
        //departure capacity
        IloNumArray D_k(env,H);
        //arrival capacity
        IloNumArray A_k(env,H);
        //sector capacity
        IloNumArray S_sector(env,j);

        //scheduled departure time
        IloNumArray D_f(env,flight);
        //scheduled arrival time
        IloNumArray R_f(env,flight);
        //turn around time after flight f;
        IloNumArray S_f(env,flight);

        // Path of a flight
        IloNumArray P(env,X+2);
        //flights' path
        IloNumArrayArray P_f(env,flight);

        // flight f cost on the ground
        IloNumArray cost_g(env,flight);
        // flight f cost in the air
        IloNumArray cost_a(env,flight);

        // Path's sector must spend time units
        IloNumArray L_sector(env,X+2);
        //flights' path-sector must spend time units
        IloNumArrayArray L_sector_f(env,flight);
        // first time period in the set T_f_j
        IloNumArray T_sector_first(env,X+2);
        // last time period in the set T_f_j
        IloNumArray T_sector_last(env,X+2);
        //flights' first time period in the set T_f_j
        IloNumArrayArray T_sector_first_f(env,flight);
        //flights' last time period in the set T_f_j
        IloNumArrayArray T_sector_last_f(env,flight);
```

object-oriented programming

input and output data(for example)

implementation procedures

Using the Cplex and C++

the problems faced when I was coding

- not familar with the Cplex and its c plus plus Reference Manual
- Promotional version problem.

convergence and computational efficiency analysis

NP hard (may not found solutions) taking the input info. into account.

scenario design

objective design

```
// TFMP final target
        IloExpr FinalRes(env);
        IloExpr FinalRes_temp1(env);
        IloExpr FinalRes_temp2(env);
        for (int n=0; n<flight; n++) {
            int Cfg=cost_g[n];
            int Cfa=cost_a[n];
            int Df=D_f[n];
            int Rf=R_f[n];
            for (int m=1; m<t; m++) {
                if(m>=T_sector_first_f[n][0]&&m<=T_sector_last_f[n][0]){
                    FinalRes_temp1+=m*(Wfst[n][0][m]-Wfst[n][0][m-1]);
                }
                if(m>=T_sector_first_f[n][X+1]&&m<=T_sector_last_f[n][X+1]){
                    FinalRes_temp2+=m*(Wfst[n][X+1][m]-Wfst[n][X+1][m-1]);
                }
            }
            FinalRes+= (Cfg-Cfa)*FinalRes_temp1+Cfa*FinalRes_temp2+(Cfa-Cfg)*Df+Cfa*Rf;

        }
```

subjective design(for example)

```
  //subject to 5
        for (int m=0; m<flight; m++) {
            for (int n=0; n<X+1; n++) {
                for (int p=0; p<t; p++) {
                    //if n~T_f_j(T_sector_first_f T_sector_last_f)
                    if(p<=T_sector_last_f[m][n]&&p>=T_sector_first_f[m][n]){
                        int lfj=L_sector_f[m][n];
                        if(p+lfj<t){
                            model.add(Wfst[m][n+1][p+lfj]<=Wfst[m][n][p]);
                        }
                    }

                }
            }
        }
```

input structure design

output analysis

case studies

For the set of 200 flights, to solve the problem CPLEX requires 234 seconds CPU time.

For the 1000, solve it at the infeasibility border over a 24 hour time period.

Compare your results to the paper.

- Paper
- mine

```
1000-0.1-(5-8)-time()-0


Found incumbent of value 197745.000000 after 0.01 sec. (2.20 ticks)
Tried aggregator 99 times.
MIP Presolve eliminated 380071 rows and 202 columns.
Aggregator did 530 substitutions.
All rows and columns eliminated.
Presolve time = 8.32 sec. (3364.02 ticks)

1000-0.1-(10-13)-time()-0

MIP Presolve eliminated 942173 rows and 834 columns.
MIP Presolve modified 97 coefficients.
Aggregator did 241 substitutions.
Reduced MIP has 144 rows, 101 columns, and 369 nonzeros.
Reduced MIP has 99 binaries, 2 generals, 0 SOSs, and 0 indicators.
Presolve time = 4.85 sec. (4314.61 ticks)
```