Celso C. Ribeiro

OR on the Ball: Applications of Combinatorial Optimization in Sports Scheduling and Management

V ALIO/EURO Workshop
Paris, October 2005

Summary

- Motivation
- Qualification problems: deciding playoffs elimination
- Timetabling:
  - Mirrored traveling tournament problem
  - Parallelization
  - Real-life application: scheduling the Brazilian soccer championship
- Referee assignment

Motivation

- Sports management is a very attractive area for original, useful and motivating applications of Operations Research.
- Competitions involve many economic and logistic issues:
  - Athlete evaluation, team evaluation, tournament planning, team management, economic estimation, marketing politics, security issues, design of fair rules, etc.
- Game scheduling involves different constraints, logistic issues, multiple objectives, and several decision makers (league officials, team managers, TV, players associations, security services, etc...).
- Many tales about unfair draws!

Applications in sports scheduling and management

Qualification:
deciding playoffs elimination

Motivation

- Press announces very early the "chances of qualification" for each team, based on estimates and statistics that are often obscure: forecasts are frequently wrong!
- Guaranteed qualification problem: how many points a team must score to be qualified (i.e., to finish in the p first positions) for the playoffs of a given tournament?
  - Win: 3 points   Tie: 1 point   Lost: 0 point
- Problem is NP-hard, but easy if each team scores only two points for a win!

Formulation

Playoffs qualification problems:

How many points a team must score to:

- ... be sure of finishing in the p first positions? (sufficient condition)
- ... be able of finishing in the p first positions? (necessary condition)

2004 edition of the Brazilian soccer championship:

- To qualify for the League of Latin America Champions: $p = 4$
- To remain in the First Division (next year): $p = 20$

Formulation

- Guaranteed qualification problem: how many points team k must score to be sure of finishing in the p first positions?
- Compute the maximum number of points this team could make and still not being qualified among the p first, then add 1 to get the minimum number of points it must score to be sure of its qualification: GQS(k)
- Possible qualification problem: how many points team k must score to have a chance of finishing in the p first positions? PQS(k)

Formulation

$$x_{ij} = \begin{cases} 1, \text{if team } i \text{ beats team } j \\ 0, \text{otherwise} \end{cases}$$

$$p_j = \text{points already accumulated by team } j$$

$$t_j = \text{points scored by team } j \text{ at the end of the tournament}$$

$$y_j = \begin{cases} 1, \text{if } t_j \geq t_k \text{ (i.e., team } k \text{ does not finish ahead of team } j) \\ 0, \text{otherwise} \end{cases}$$

Formulation

$$GQS(k) = 1 + \max t_k$$

k = team being considered

subject to :

$$x_{ij} + x_{ji} \leq 1 \quad \forall i = 1,\ldots,n; \forall j = 1,\ldots,n : \text{game } (i,j) \text{ still to be played}$$

$$t_j = p_j + 3.\sum_{i \neq j} x_{ji} + \sum_{i \neq j} \left[1 - (x_{ij} + x_{ji})\right] \quad \forall j = 1,\ldots,n$$

$$t_k - t_j \leq M(1 - y_j) \quad \forall j = 1,\ldots,n$$

$t_k > t_j \Rightarrow y_j = 0$

$$\sum_{j \neq k} y_j \geq p$$

$$x_{ij} \in \{0,1\} \quad \forall i = 1,\ldots,n; \forall j = 1,\ldots,n : \text{game } (i,j) \text{ still to be played}$$

$$y_j \in \{0,1\} \quad \forall j = 1,\ldots,n$$

Formulation

- Ties in the number of points may be resolved in favor of teams with more wins.
- In this model, add a very small value (necessarily smaller than one) to the number of points:

$$t_j = p_j + (3+\varepsilon)\sum_{i\neq j} x_{ji} + \sum_{i\neq j}[1-(x_{ij}+x_{ji})]$$

- Use, e.g., $\varepsilon = 0.01$
- Using a similar model, compute PQS(k): minimum number of points for possible classification

Formulation

- When is team k considered as "mathematically qualified"?

  Team k is mathematically qualified when the previous problem is infeasible.

- When does team k depend only on itself to be qualified?

  Team k depends only on itself if GQS(k) is less than or equal to the total number of points it still has to play (all wins).

- When is team k considered as "mathematically eliminated"?

  Team k is mathematically eliminated when the associated possible qualification problem is infeasible.

FUTMAX in the Net

- FUTMAX project:
  - Multiagent system collects results from the web.
  - Integer programming models generated for all teams.
  - Models solved by CPLEX.
  - HTML file automatically created from results.
- Automatic publication of the results:

  http://www.futmax.org (L.A. league of champions)

FUTMAX in the press

- Articles in journals, radio, and TV.
- SPORTV broadcasts statistics provided by FUTMAX.
- Spin-offs: HockeyPlex project (same idea applied to the National Hockey League, USA)
- FUTMAX was used several times to disprove statements made by team managers and the press.

FUTMAX in the press

Notícias

- Em cima da hora
- Brasil
- Mundo
- Dinheiro
- Cotidiano
- Esporte
- Ilustrada
- Informática
- Ciência
- Educação
- Galeria de imagens
- Especiais

Serviços

31/10/2002 - 22h19

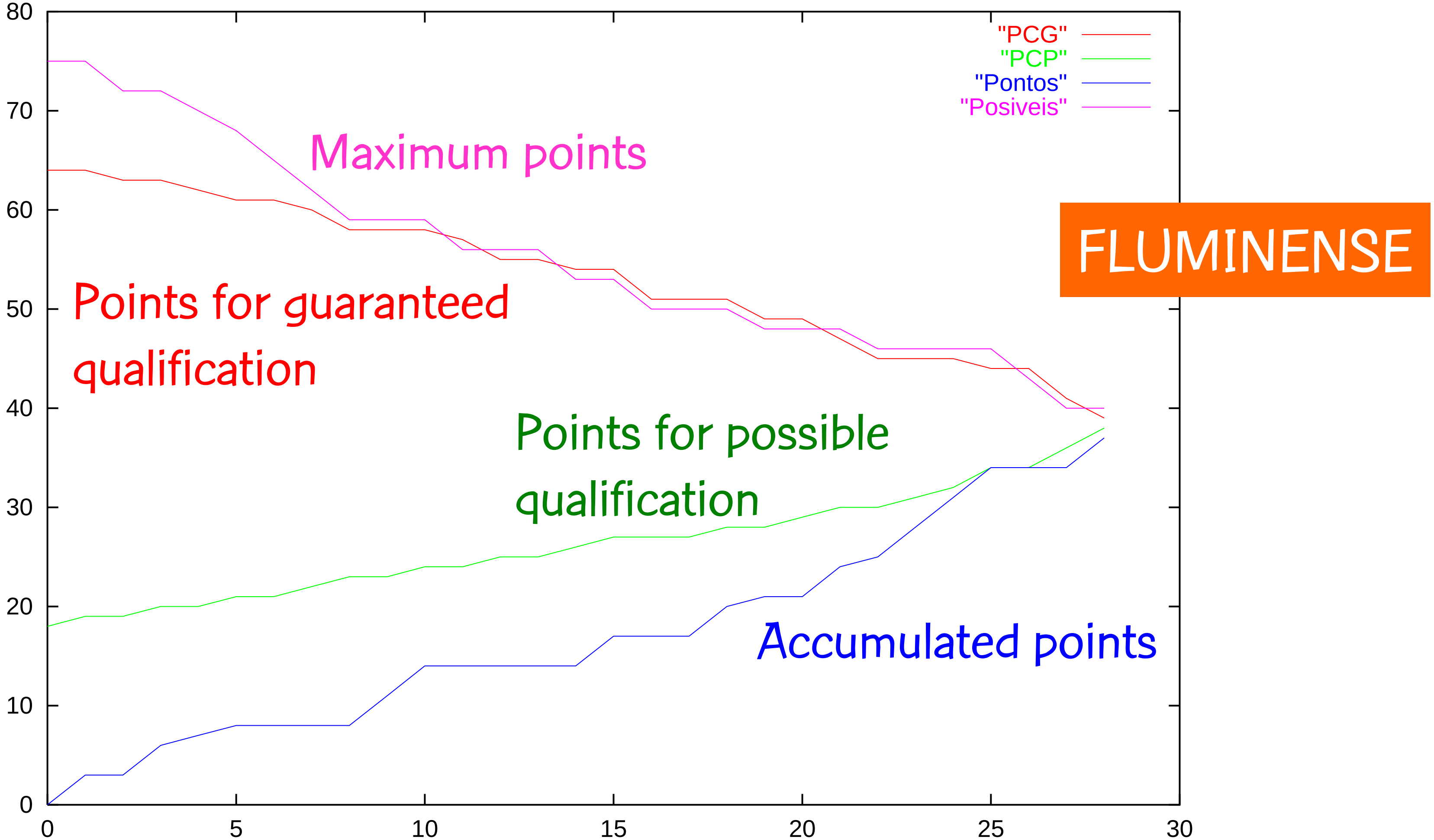
São Paulo bate a Ponte de virada, dispara na liderança e se classifica

da **Folha Online**

Líder isolado, único time classificado para a próxima fase, com maior número de vitórias (13 em 23 jogos), [illegible] e o artilheiro do [illegible] São Paulo encerrou a rodada do Campeonato Brasileiro, com uma vitória por 5 a 2 sobre a Ponte Preta, na noite de hoje, no estádio do Morumbi.

Results

FUTMAX may be used to address the behavior of each team:



Publications

- Ribeiro & Urrutia, "OR on the ball", *OR/MS Today*, 2004.
- Ribeiro & Urrutia, "An application of integer programming to playoff elimination in football championships", *International Transactions in OR*, 2005.
- Noronha, Lucena, Ribeiro, & Urrutia, "A multi-agent framework to retrieve and publish information about qualification and elimination in sports championships", 2005.

Applications in sports scheduling and management

Timetabling: mirrored traveling tournament problem

Motivation

- Game scheduling involves different constraints, logistic issues, multiple objectives, and several decision makers (league officials, team managers, TV stations, players' associations, security services, etc.).
- Total distance to be traveled is one of the important variables to be optimized, to reduce traveling costs and to give more time to the players for resting and training: unfair schedules should be avoided.
- Timetabling is the major area of applications of OR in sports.

Problem formulation

- Conditions:
  - n (even) teams take part in a tournament.
  - Each team has its own stadium at its home city.
  - Distances between the stadiums are known.
  - A team playing two consecutive away games goes directly from one city to the other, without returning to its home city.

Problem formulation

- Conditions:
  - Compact double round-robin tournament:
    - There are 2(n-1) rounds, each one with n/2 games.
    - Each team plays against every other team twice, once at home and once away.
  - No team can play more than three games in a home stand (home games) or in a road trip (away games).
- Traveling tournament problem (TTP): minimize the total distance traveled by all teams.
- Complexity: still open!
- Largest instance solved to date (sequential): n=6

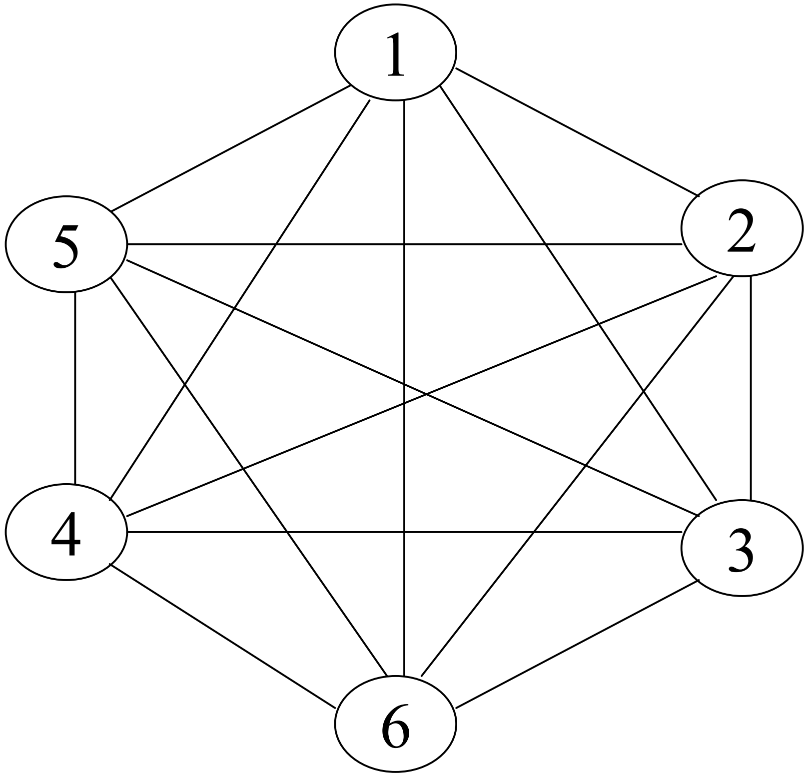
Problem formulation

- Mirrored Traveling Tournament Problem (MTTP):
  - All teams face each other once in the first phase with the first n-1 rounds.
  - In the second phase with the last n-1 rounds, the teams play each other again in the same order, following an inverted home/away pattern.
  - Common structure in Latin-American tournaments.
- Set of feasible solutions to the MTTP is a subset of the feasible solutions to the TTP.

1-factorizations

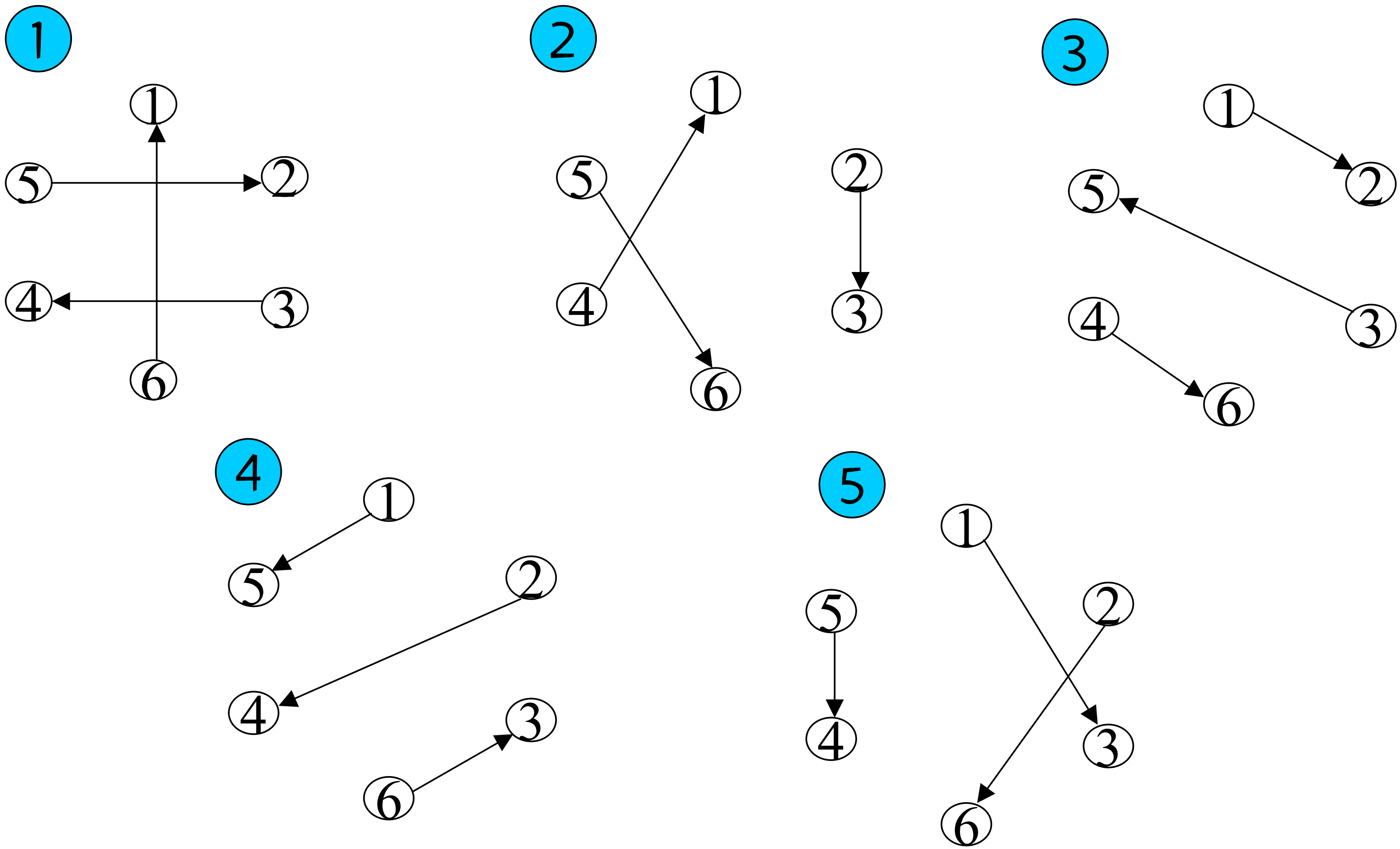
- Given a graph $G=(V, E)$, a factor of G is a graph $G'=(V,E')$ with $E'\subseteq E$.
- G' is an 1-factor if all its nodes have degree equal to one.
- An 1-factorization of $G=(V,E)$ is a set of edge-disjoint 1-factors $G^1=(V,E^1)$, ..., $G^p=(V,E^p)$, such that $E^1\cup\ldots\cup E^p=E$.
- Oriented 1-factorization: assign orientations to the edges of an 1-factorization.
- Ordered oriented 1-factorization: assign an order to the factors of an oriented 1-factorization.

1-factorizations



Example:
Ordered oriented
1-factorization of K_6

1-factorizations



1-factorizations

- Mirrored tournament: games in the second phase are determined by those in the first
  - Each edge of K_n represents a game.
  - Each 1-factor of K_n represents a round.
  - Each ordered oriented 1-factorization of K_n represents a schedule for n teams.
- Problem is huge: there are 526,915,620 non-isomorphic 1-factorizations of K_{12} (Dinitz, Garnick, & McKay, 1995)

Constructive heuristic

- Initial solution: three phases
  1. Schedule games using abstract teams: polygon method defines the structure of the tournament, which will not change in the other steps.
  2. Assign real teams to abstract teams: greedy heuristic to QAP (number of travels between stadiums of the abstract teams x distances between the stadiums of the real teams).
  3. Select stadium for each game (home/away pattern) in the first phase (mirrored tournament): random assignment in the first round, simple rules, local search using home-away swaps.

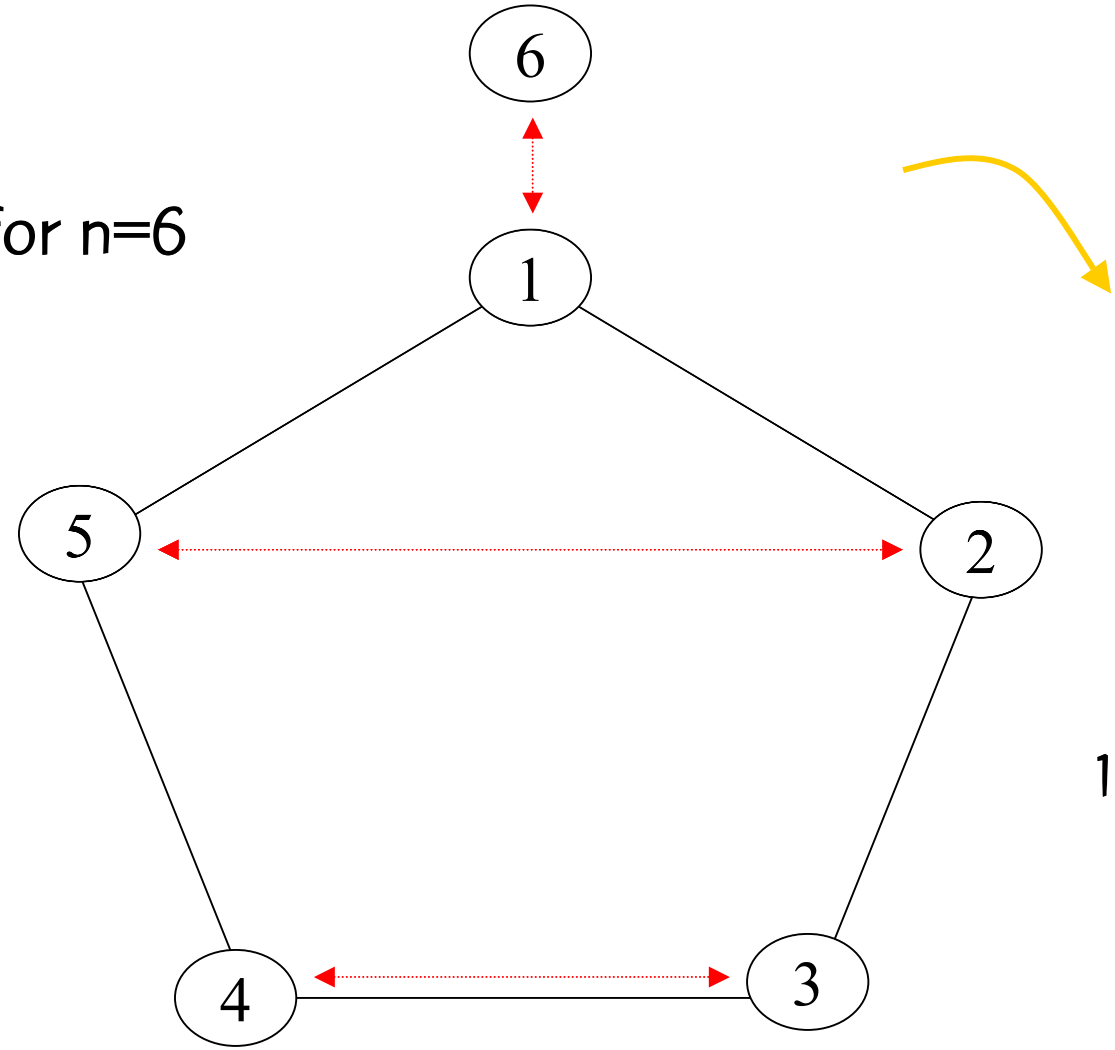
Constructive heuristic

- Phase 1:
  - Schedule games using abstract teams: polygon method defines the structure of the tournament.
  - Tornament structure will not change in the other phases.

Constructive heuristic

Example:

polygon method for n=6

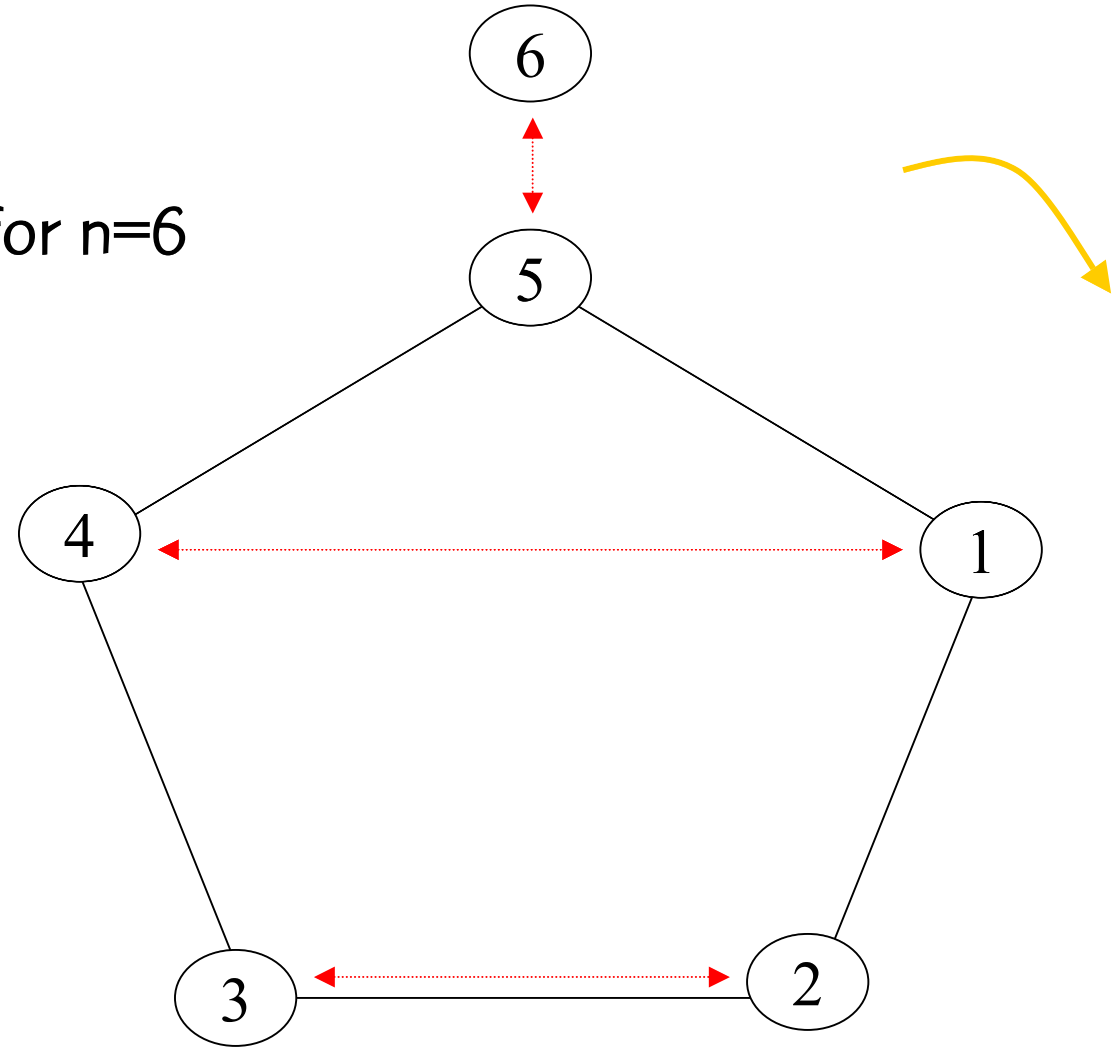


1^{st} round

Constructive heuristic

Example:

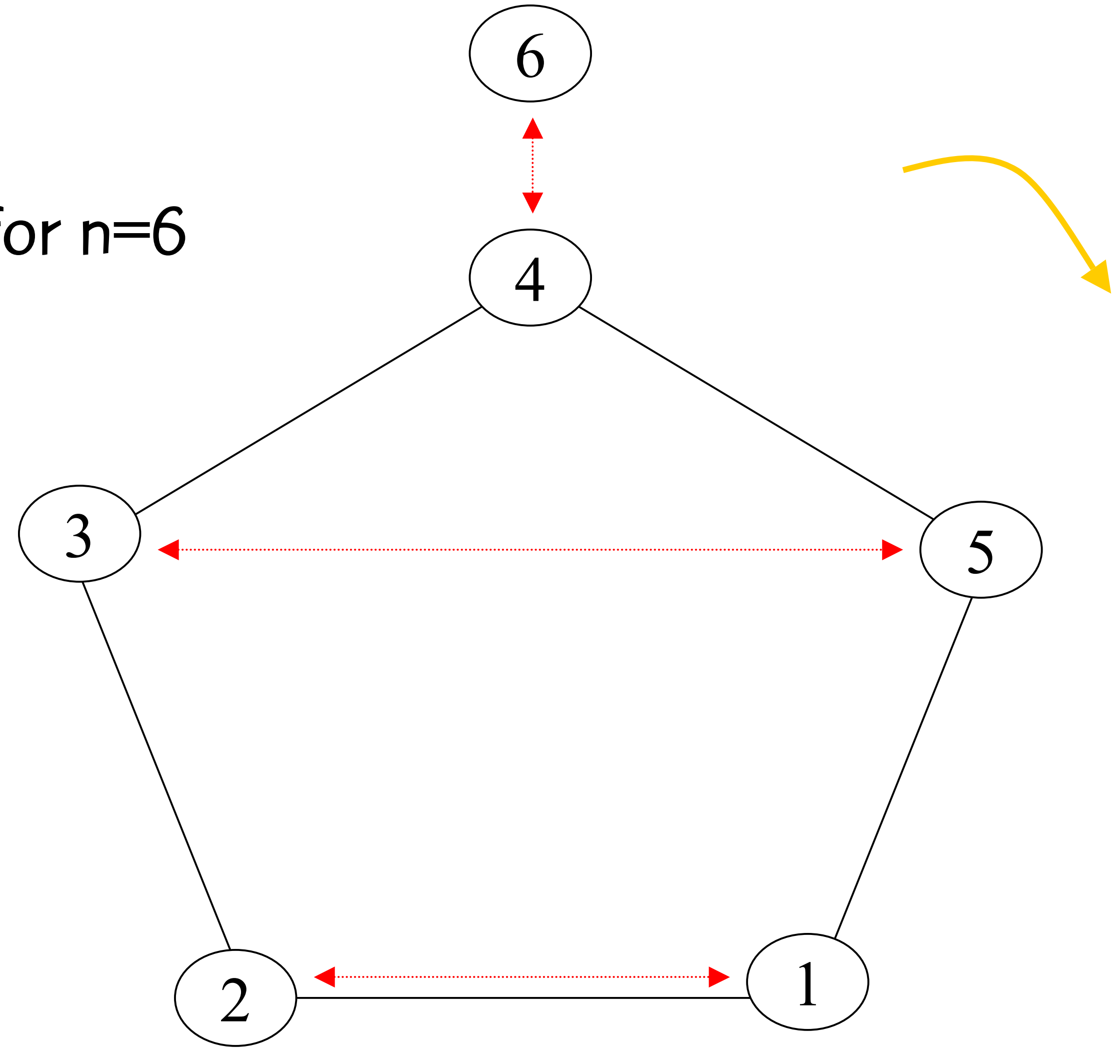
polygon method for n=6



2nd round

Constructive heuristic

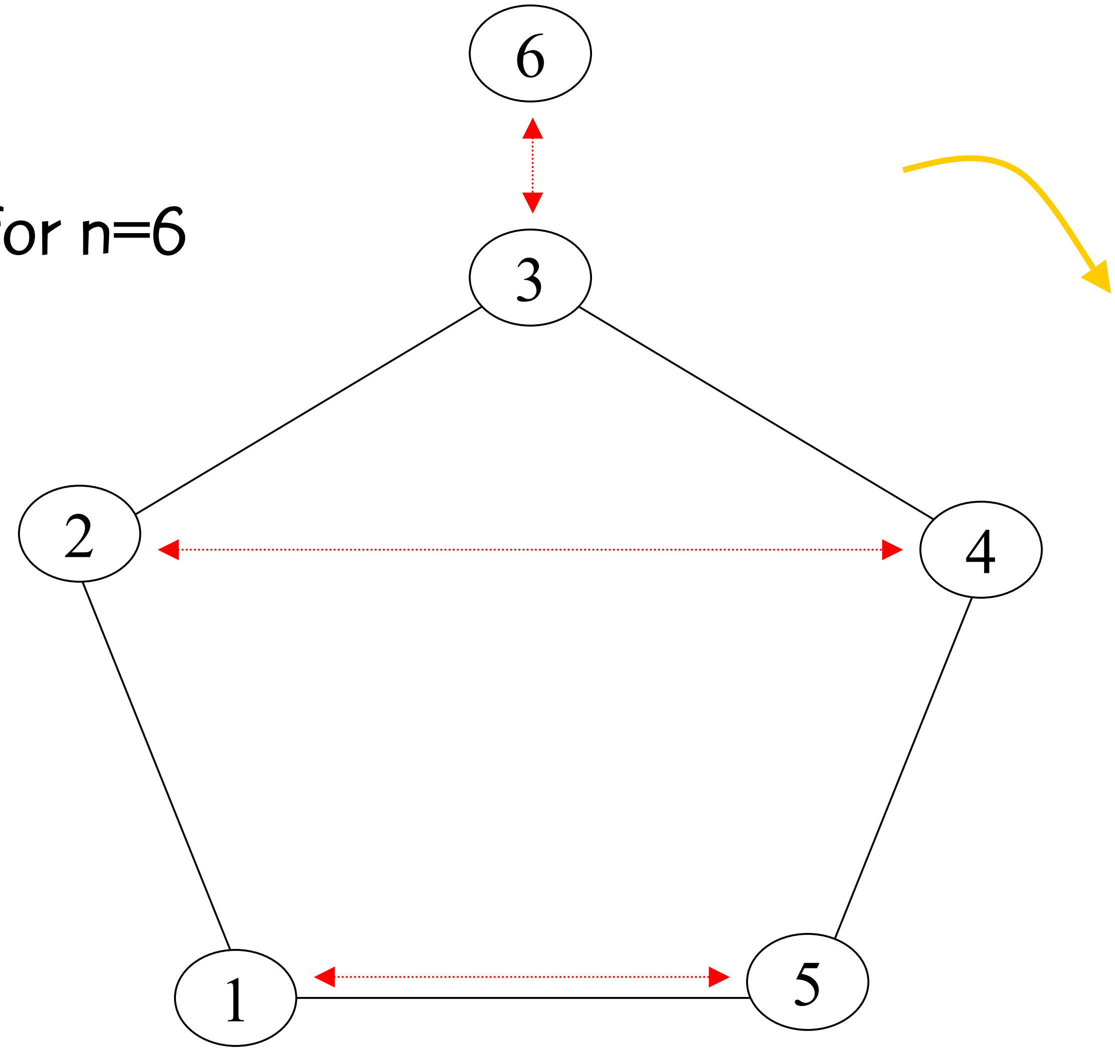
Example:
polygon method for n=6



3rd round

Constructive heuristic

Example:
polygon method for n=6

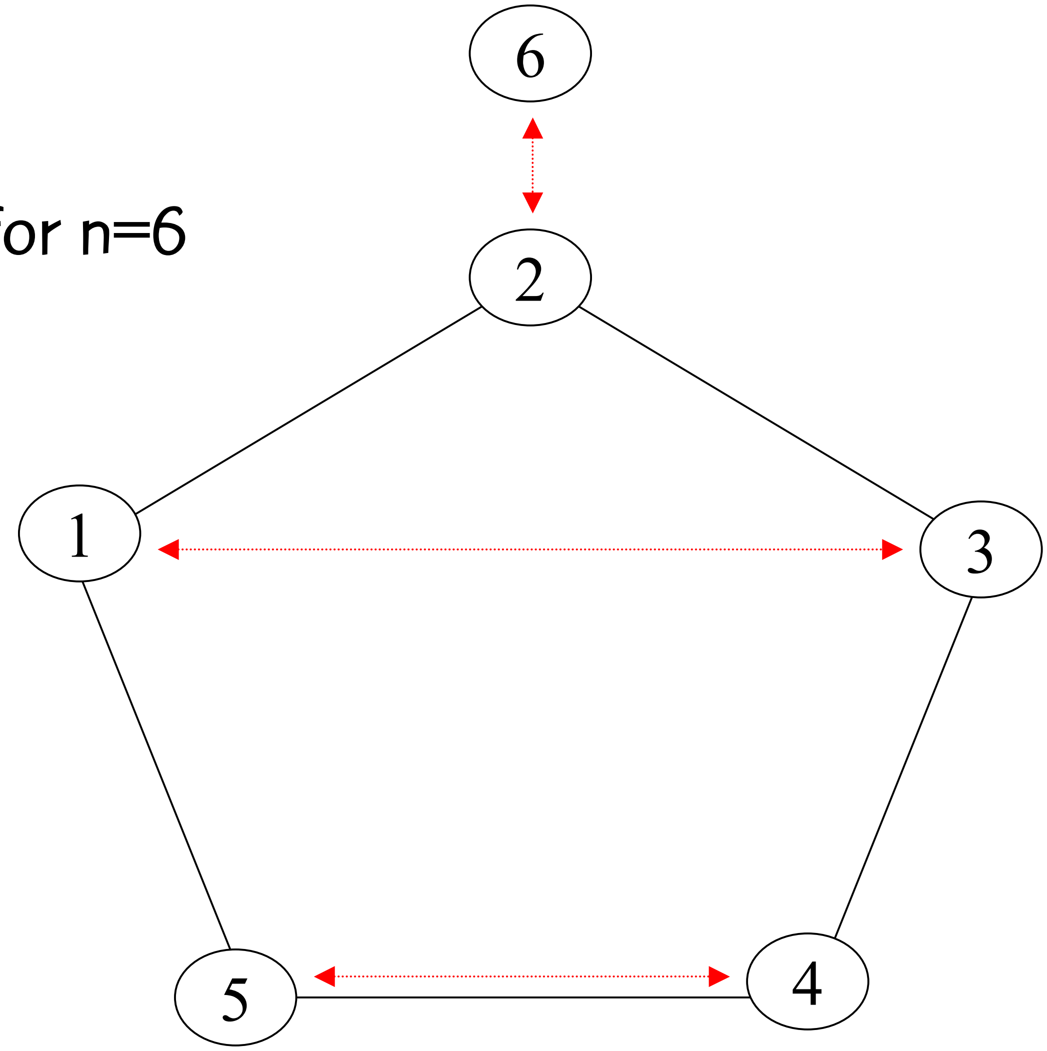


4^{th} round

Constructive heuristic

Example:

polygon method for n=6



5^{th} round

Constructive heuristic

- Phase 2: assign real teams to abstract teams
  - Build a matrix with the number of consecutive games for each pair of abstract teams X and Y: the entry corresponding to X and Y gives the total number of times another team plays consecutively with X and Y, in any order.
  - Assign real teams to abstract teams using a greedy algorithm: pairs of real teams whose stadiums are close are assigned to pairs of abstract teams with large entries in the matrix of consecutive games (QAP heuristic).

Constructive heuristic

	A	B	C	D	E	F
A	0	1	6	5	2	4
B	1	0	2	5	6	4
C	6	2	0	2	5	3
D	5	5	2	0	2	4
E	2	6	5	2	0	3
F	4	4	3	4	3	0

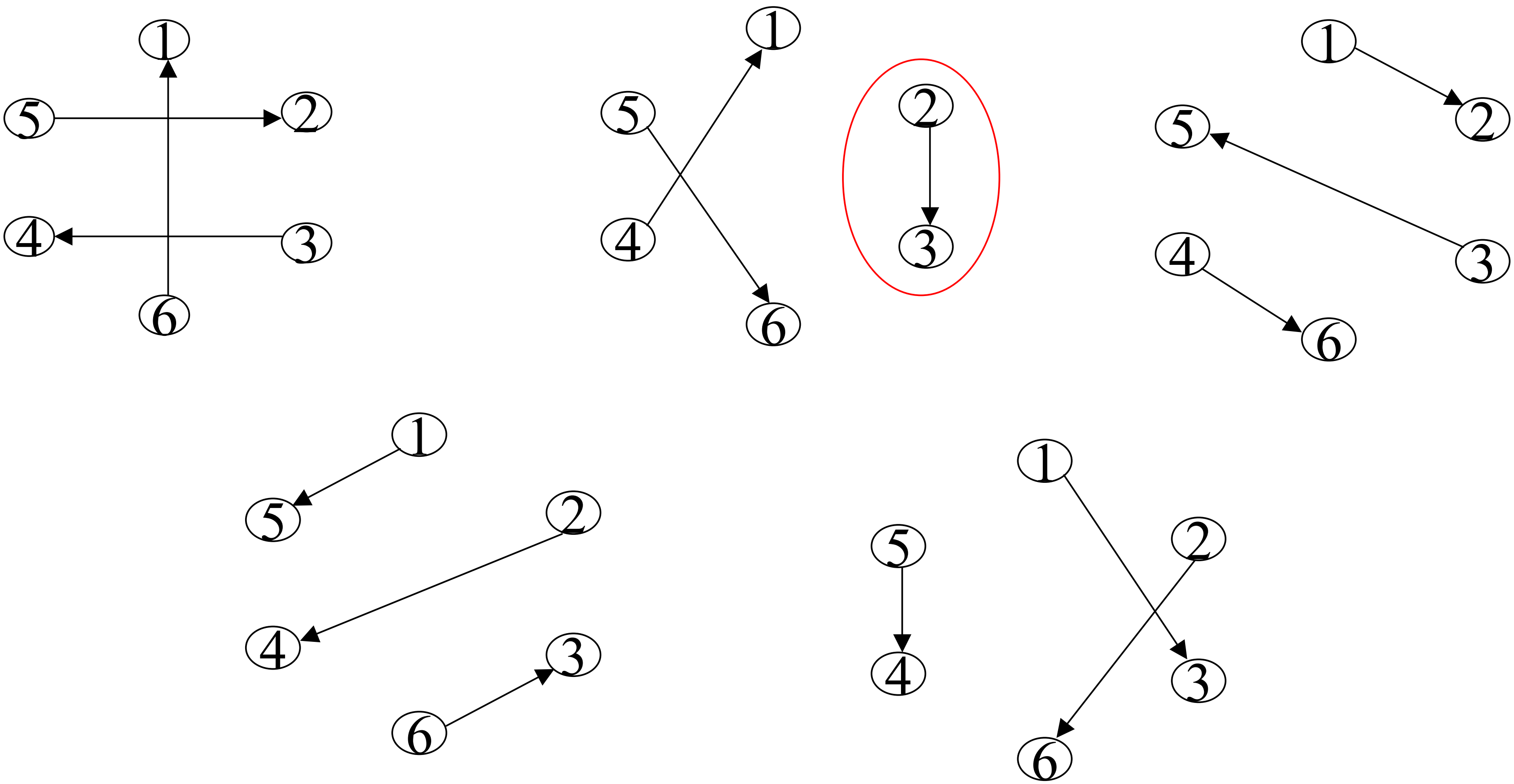
Constructive heuristic

- Phase 3: assign stadiums to games
  - Build a feasible assignment of stadiums, starting from a random assignment in the first round.
  - Improve this assignment, using a simple local search algorithm based on home-away swaps.

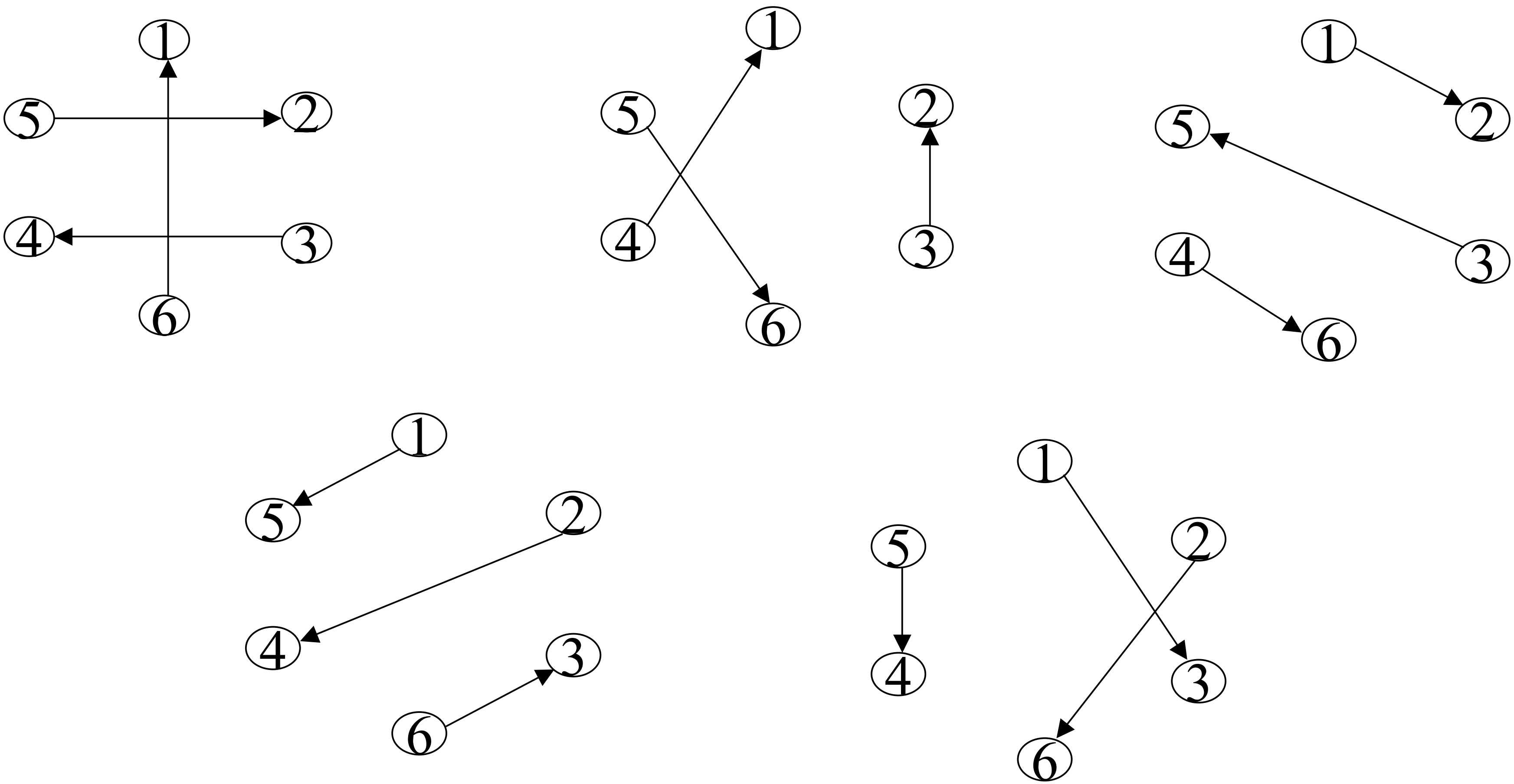
Local search

- Metaheuristics based on local search.
- Local search explores moves in three neighborhoods:
  - Home-away swaps: change the stadium of a single game
  - Team swaps: exchange the games of two teams over all rounds
  - Partial round swaps: find two games AxB and CxD in round X and two games AxC and BxD in round Y and exchange the rounds where they take place (only for n≥8, not always exist).
- Moves in these neighborhoods are easy to be computed, but not always lead to good hidden solutions.

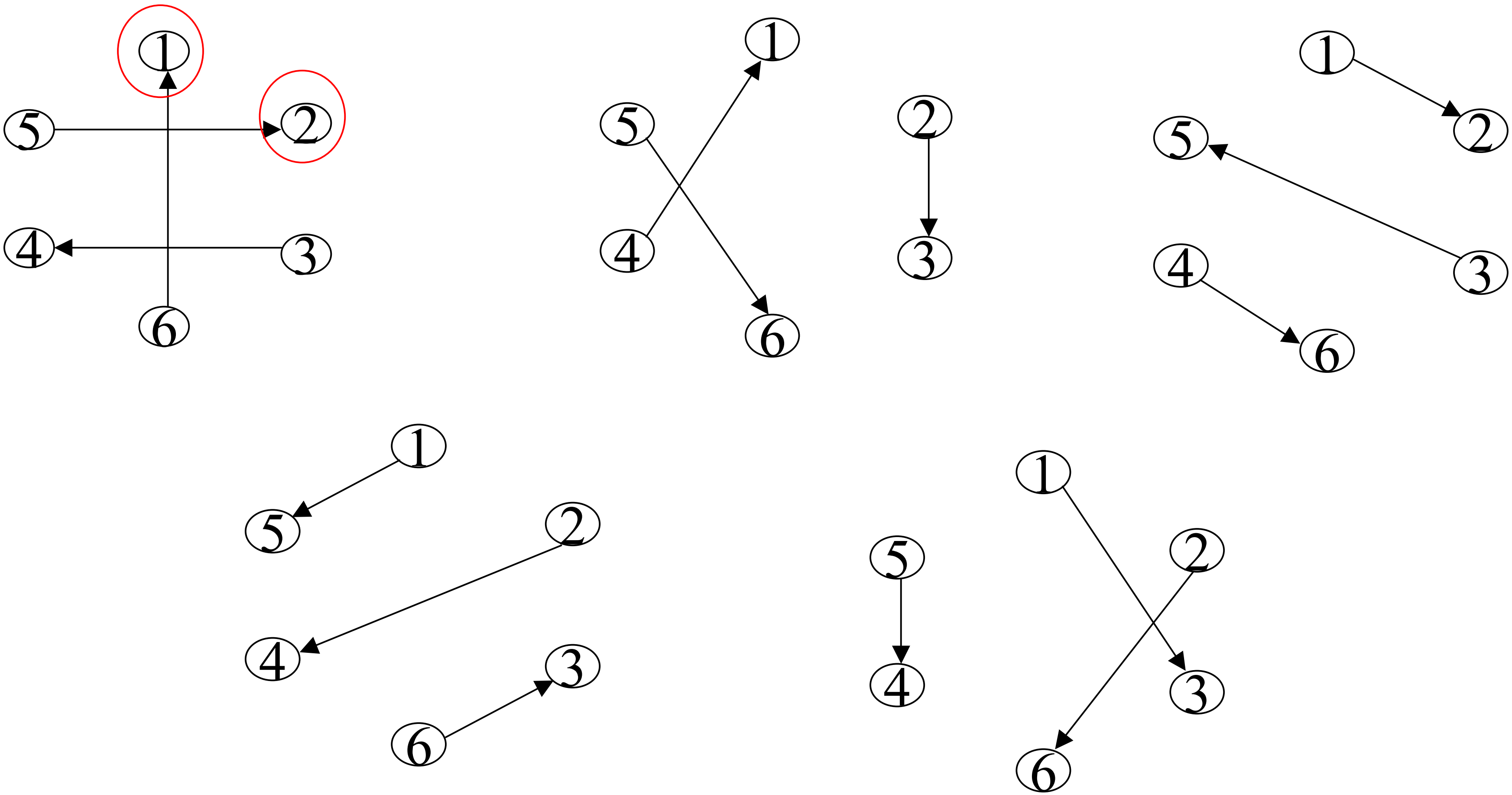
Neighborhood: home-away swap



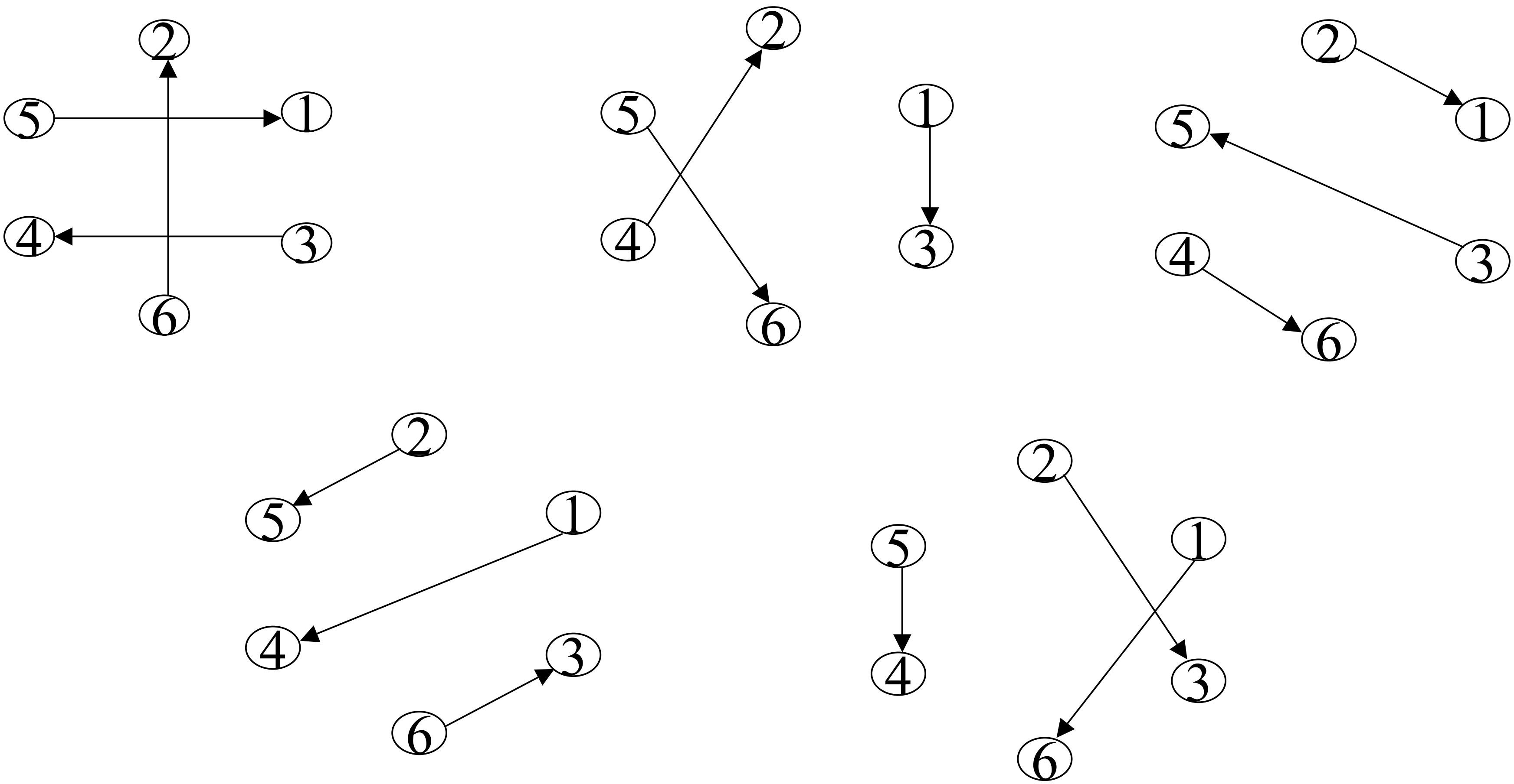
Neighborhood: home-away swap



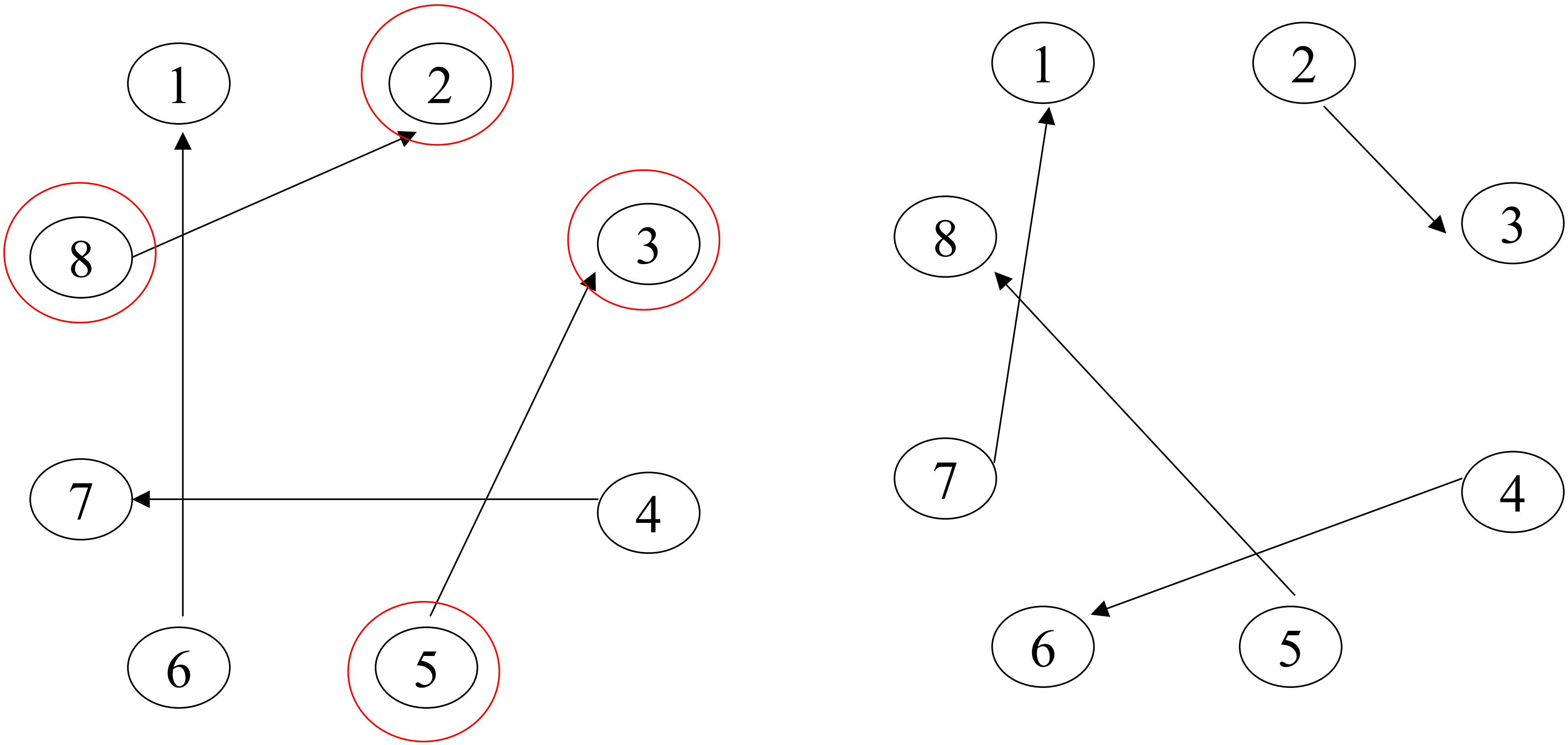
Neighborhood: team-swap



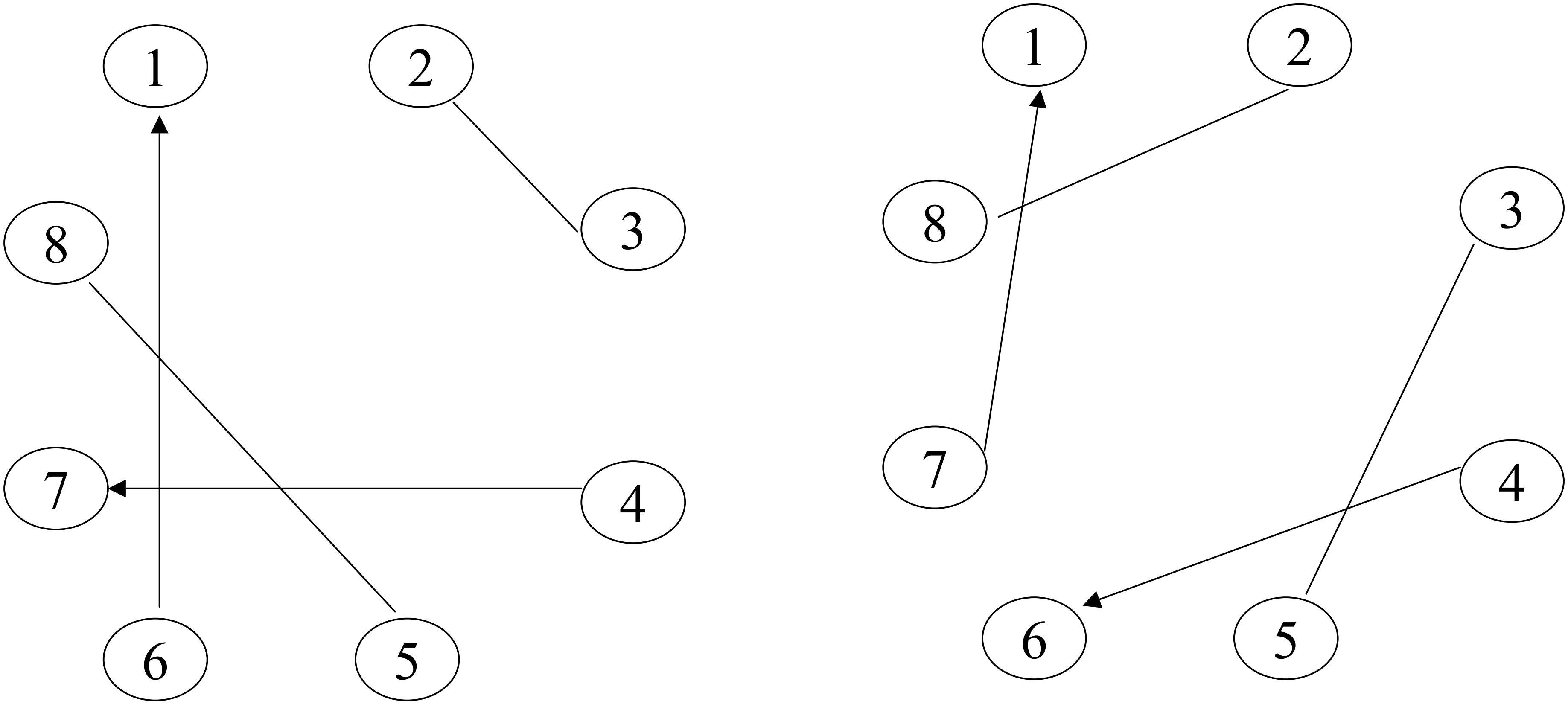
Neighborhood: team-swap



Neighborhood: partial round swap



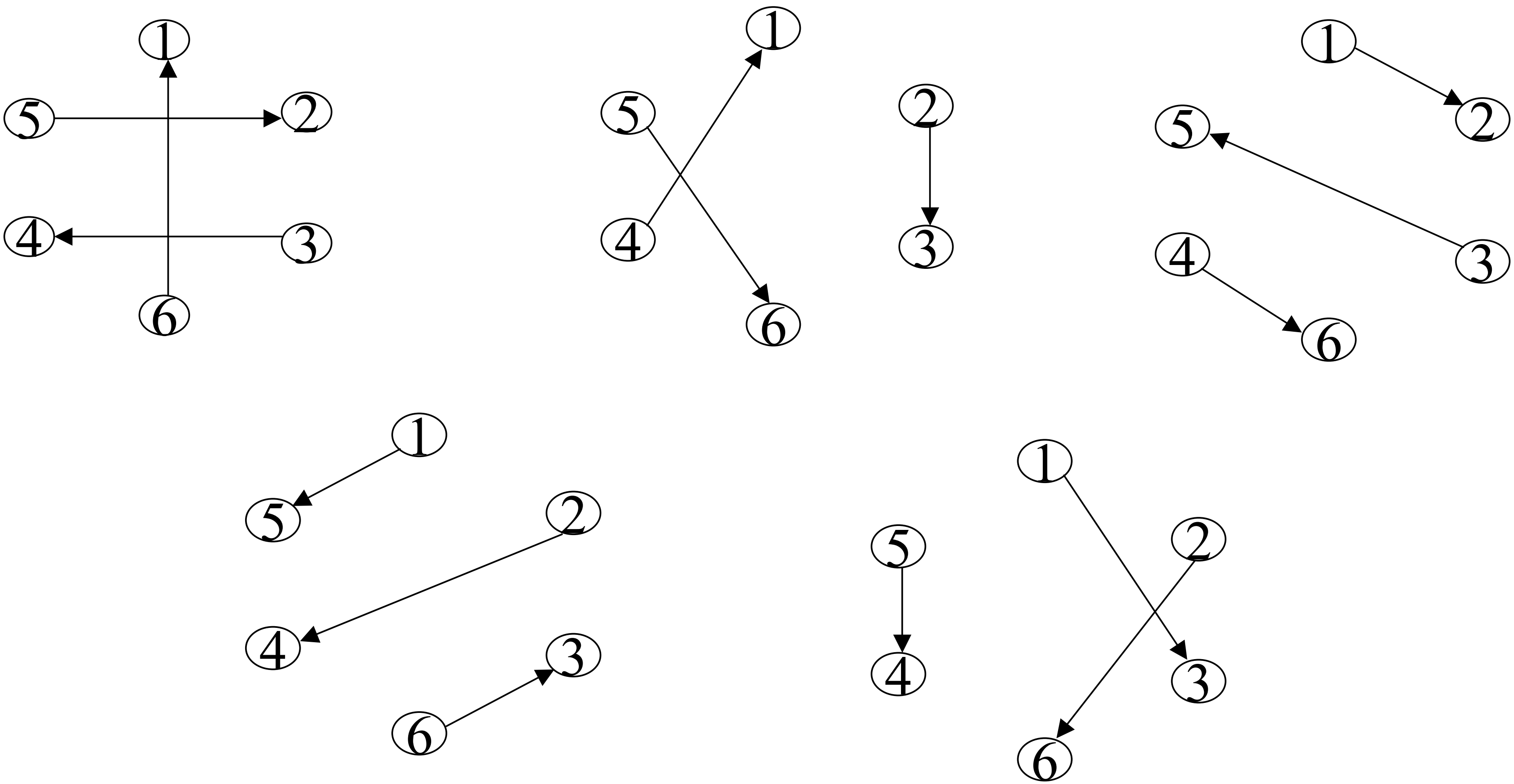
Neighborhood: partial round swap



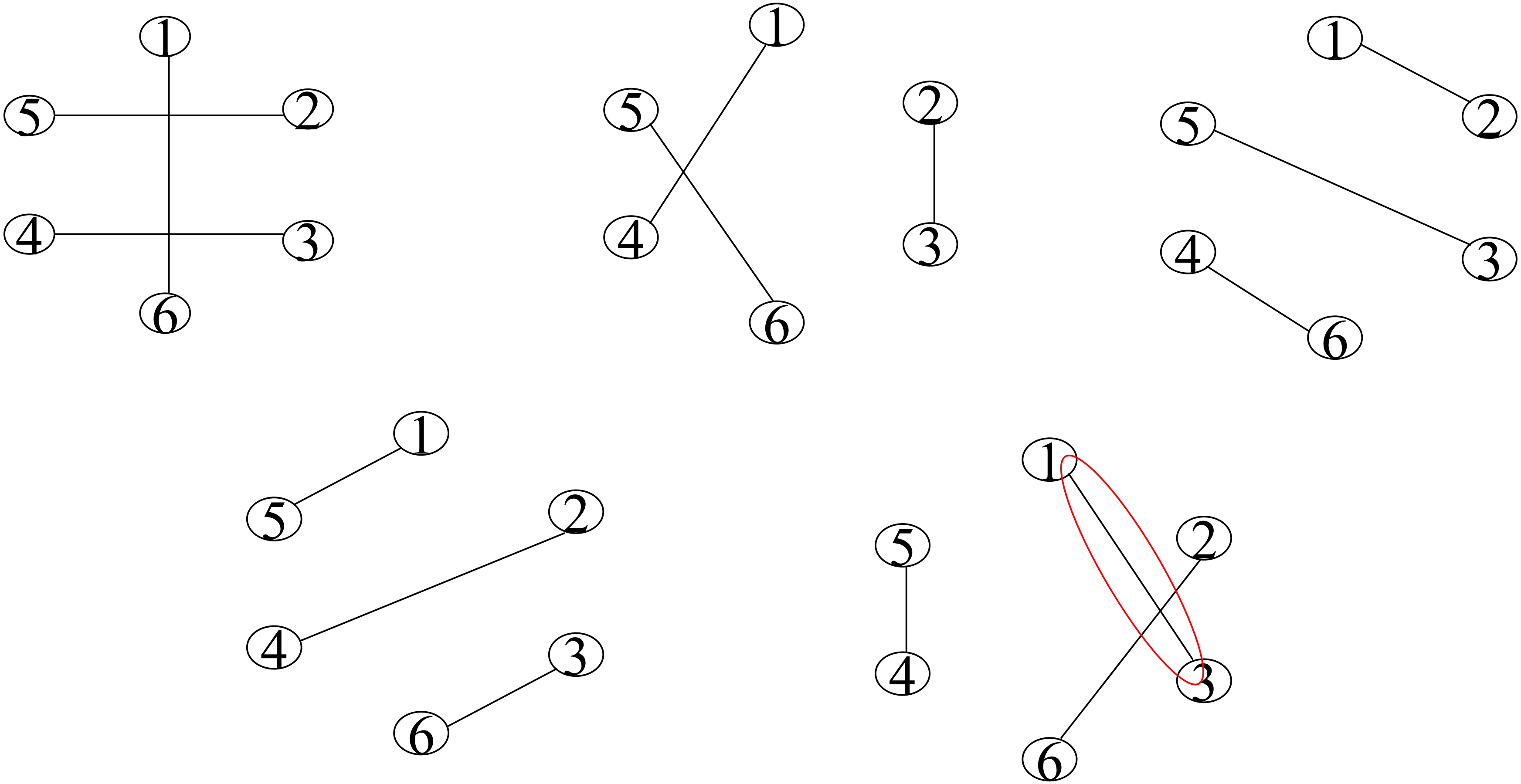
Local search

- Moves in a more complex neighborhood are explored as perturbations.
- Ejection chain:
  - Enforce a game into a given round.
  - Use an ejection chain to recover an 1-factorization.

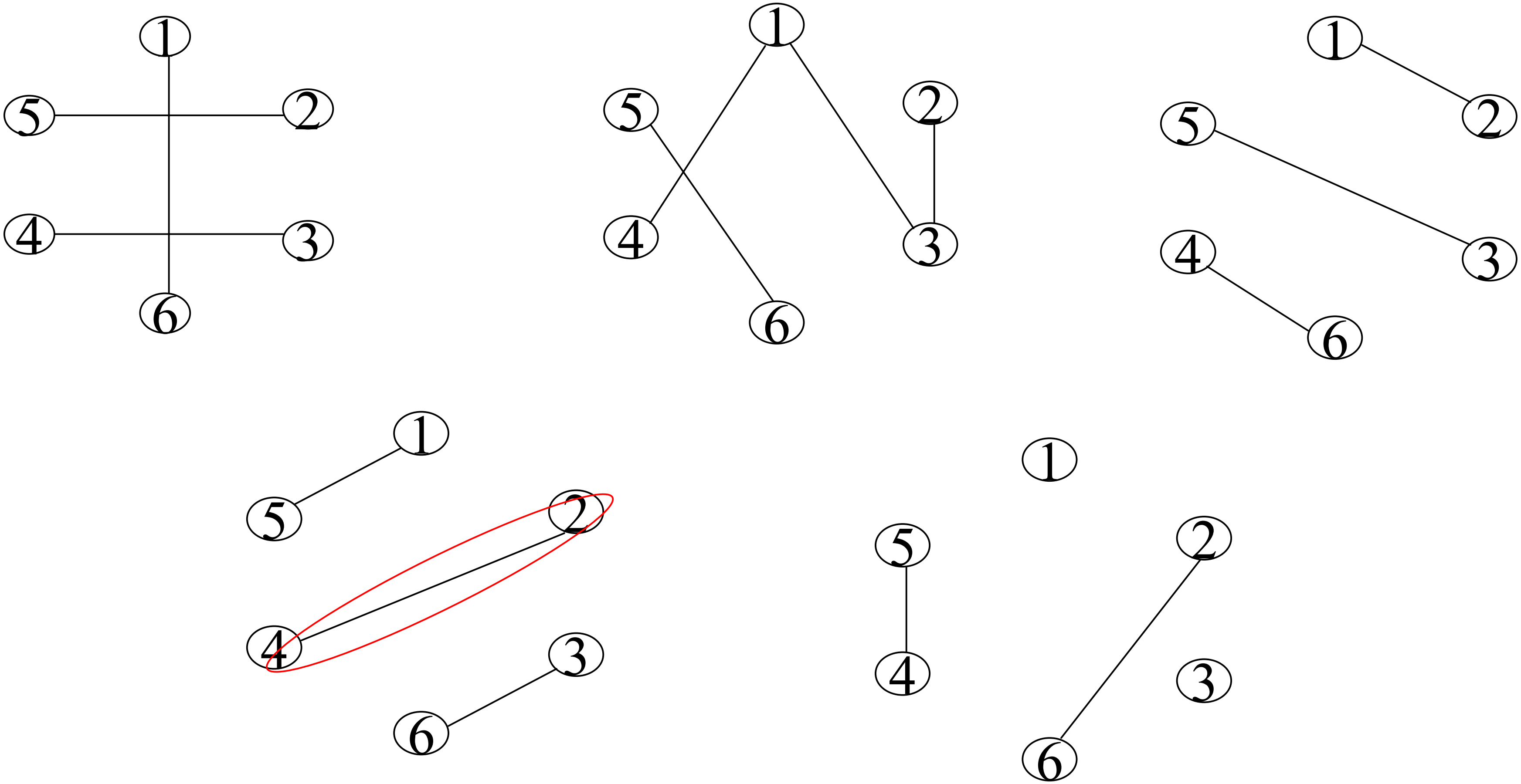
Ejection chain: game rotation



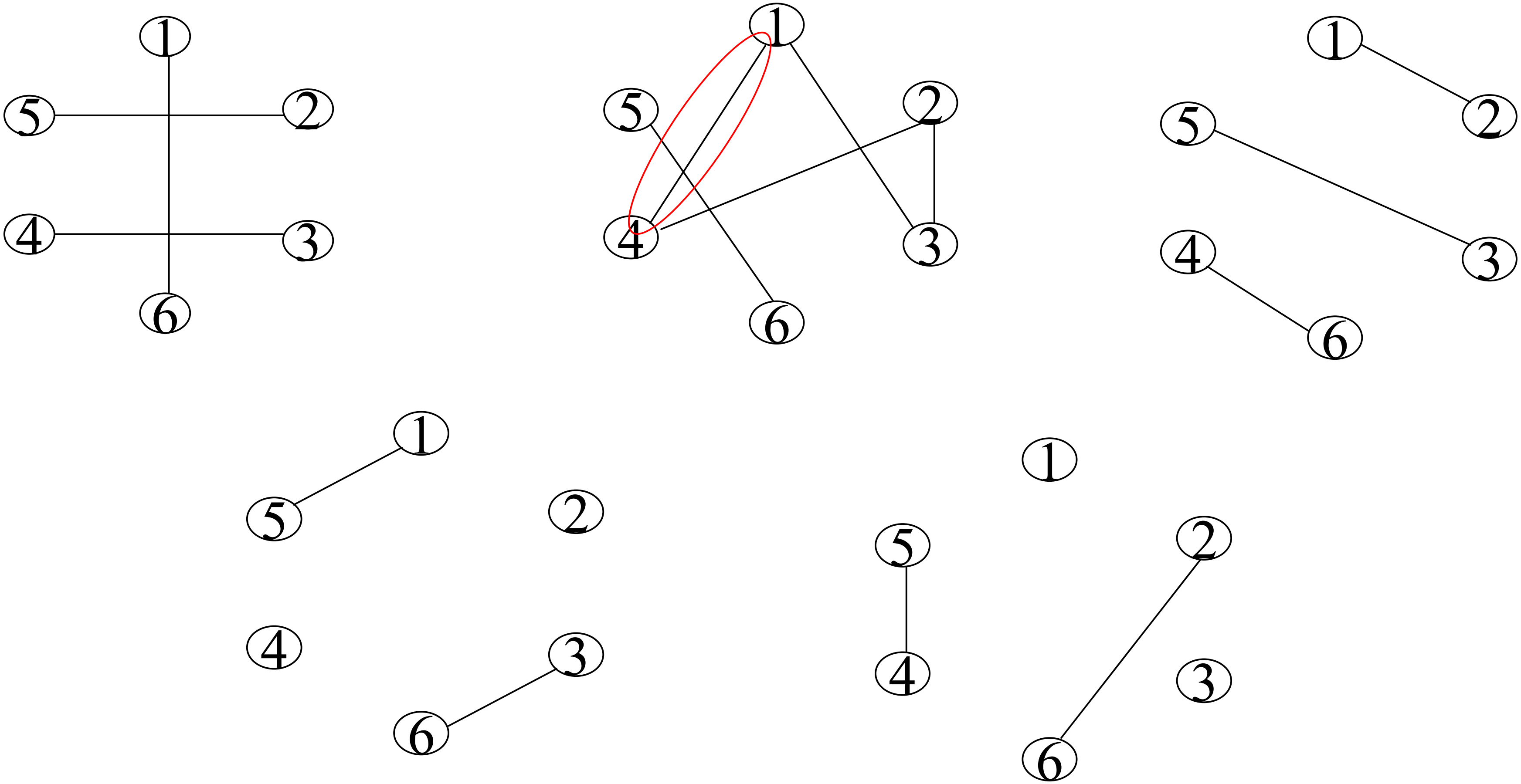
Ejection chain: game rotation



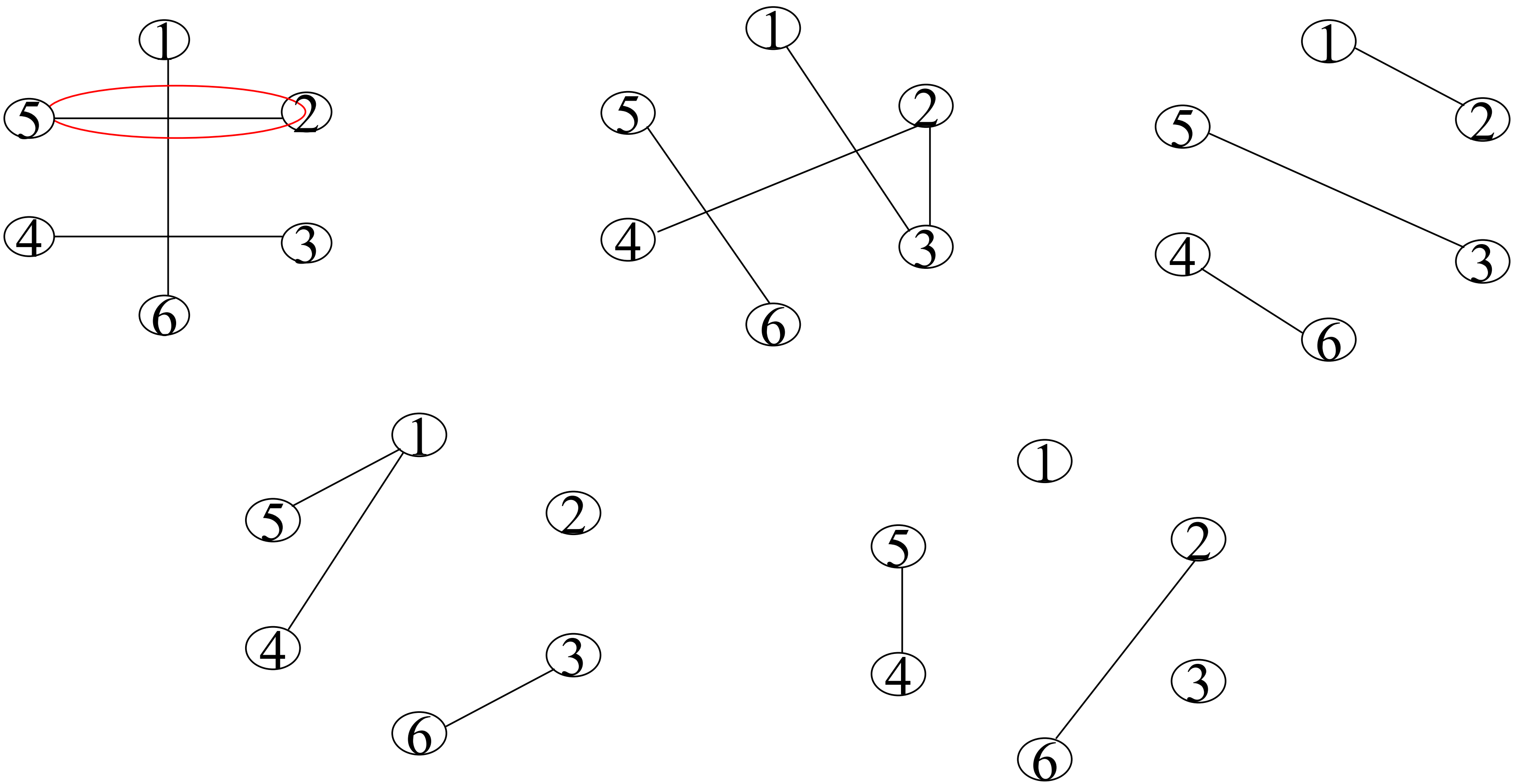
Ejection chain: game rotation



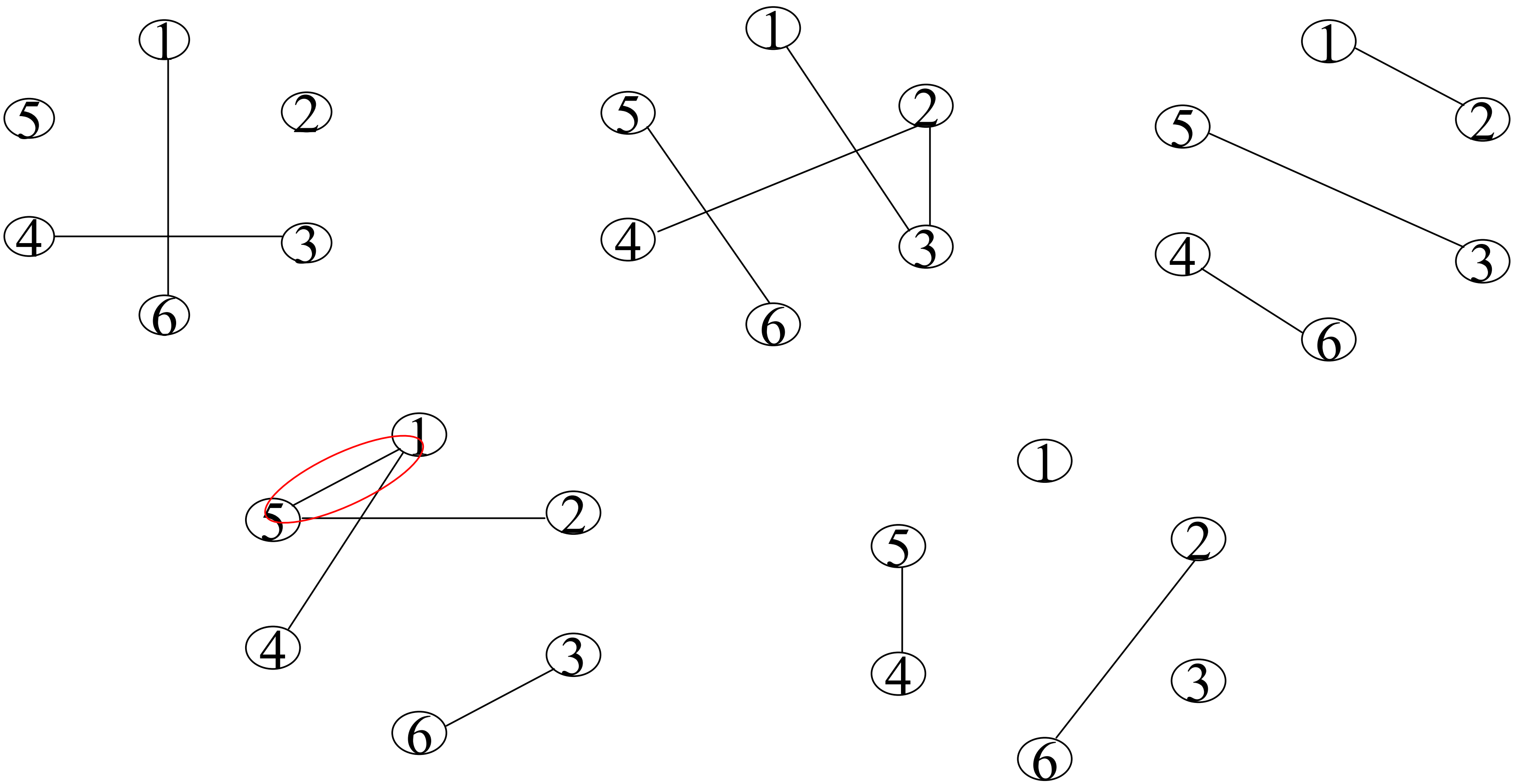
Ejection chain: game rotation



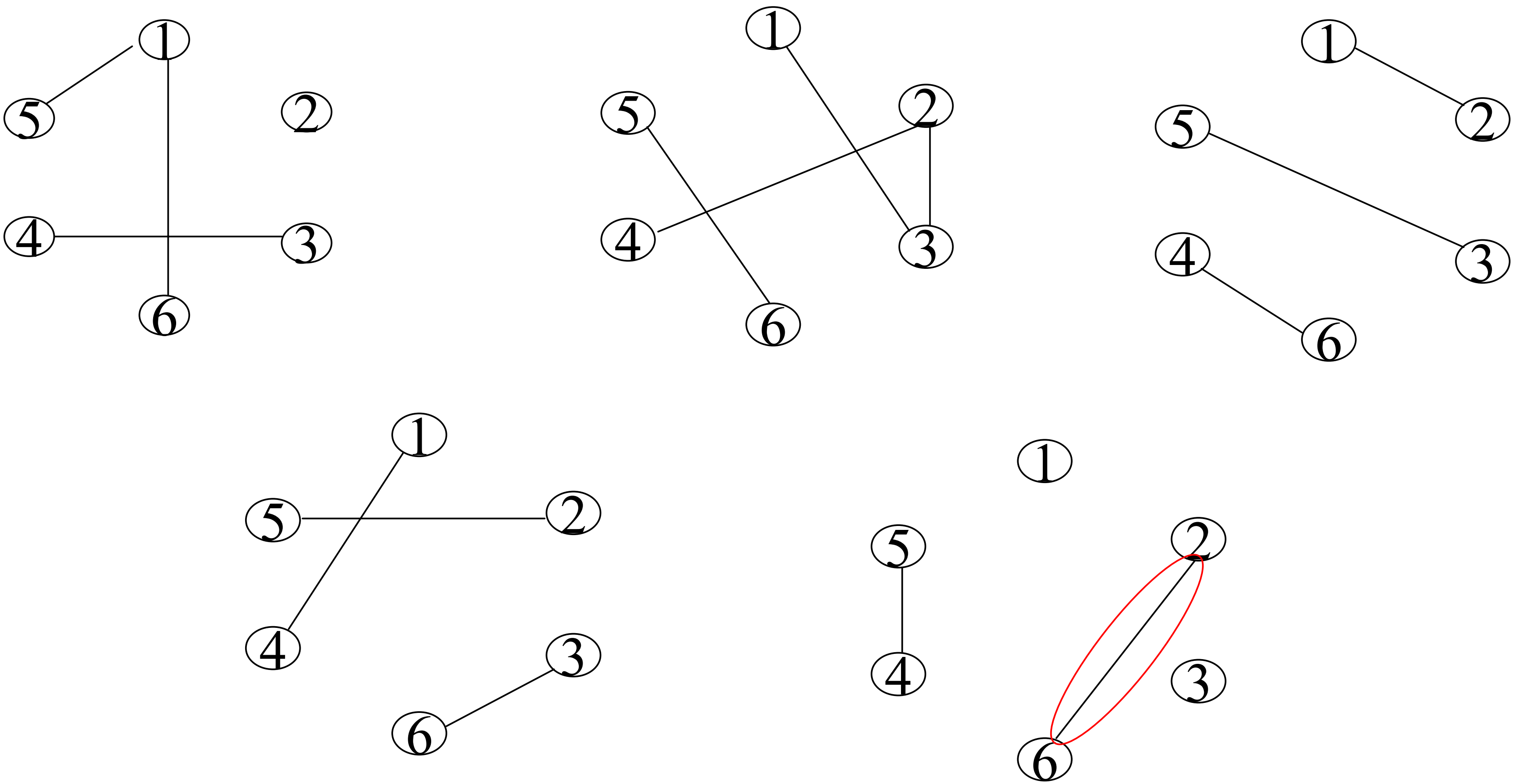
Ejection chain: game rotation



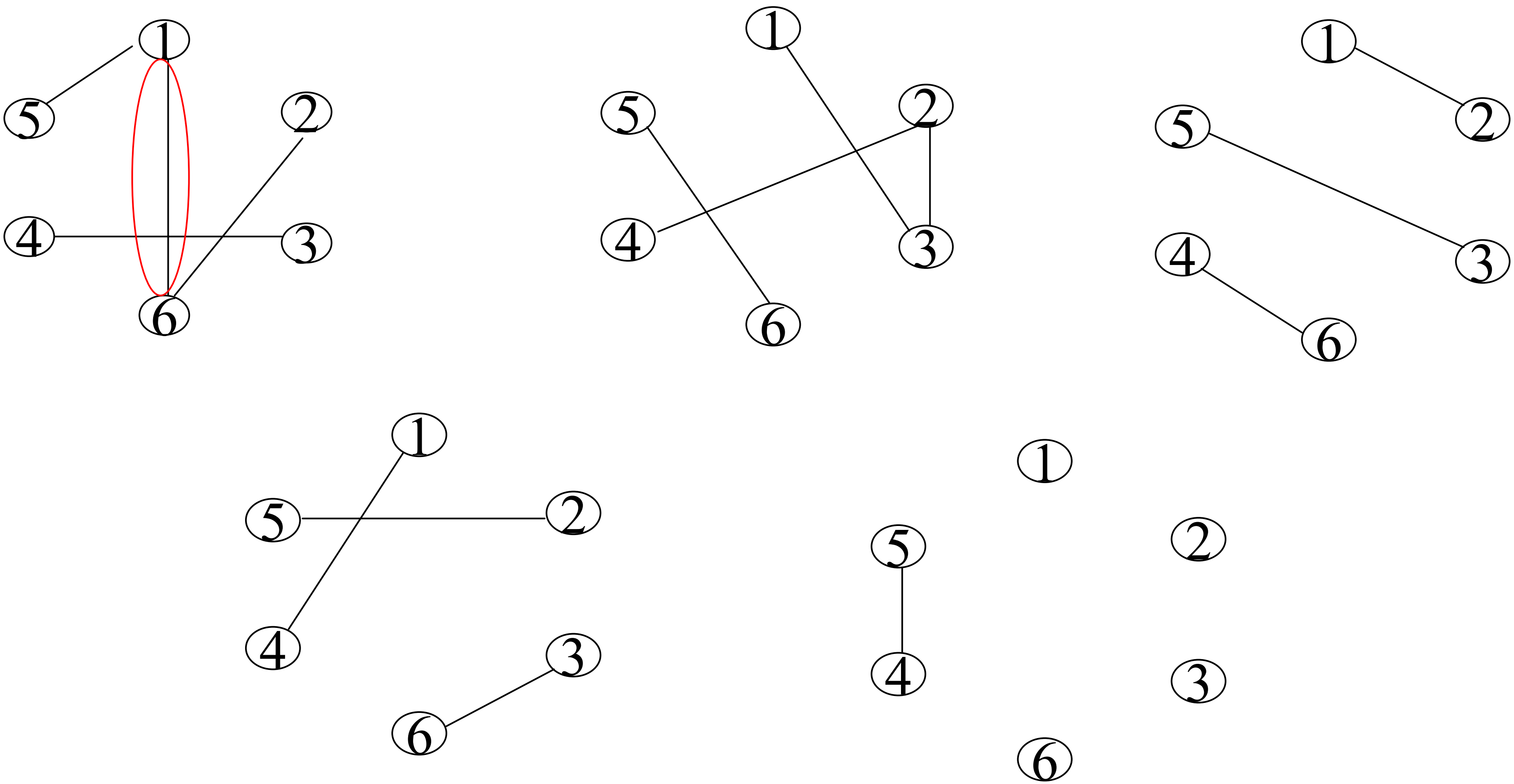
Ejection chain: game rotation



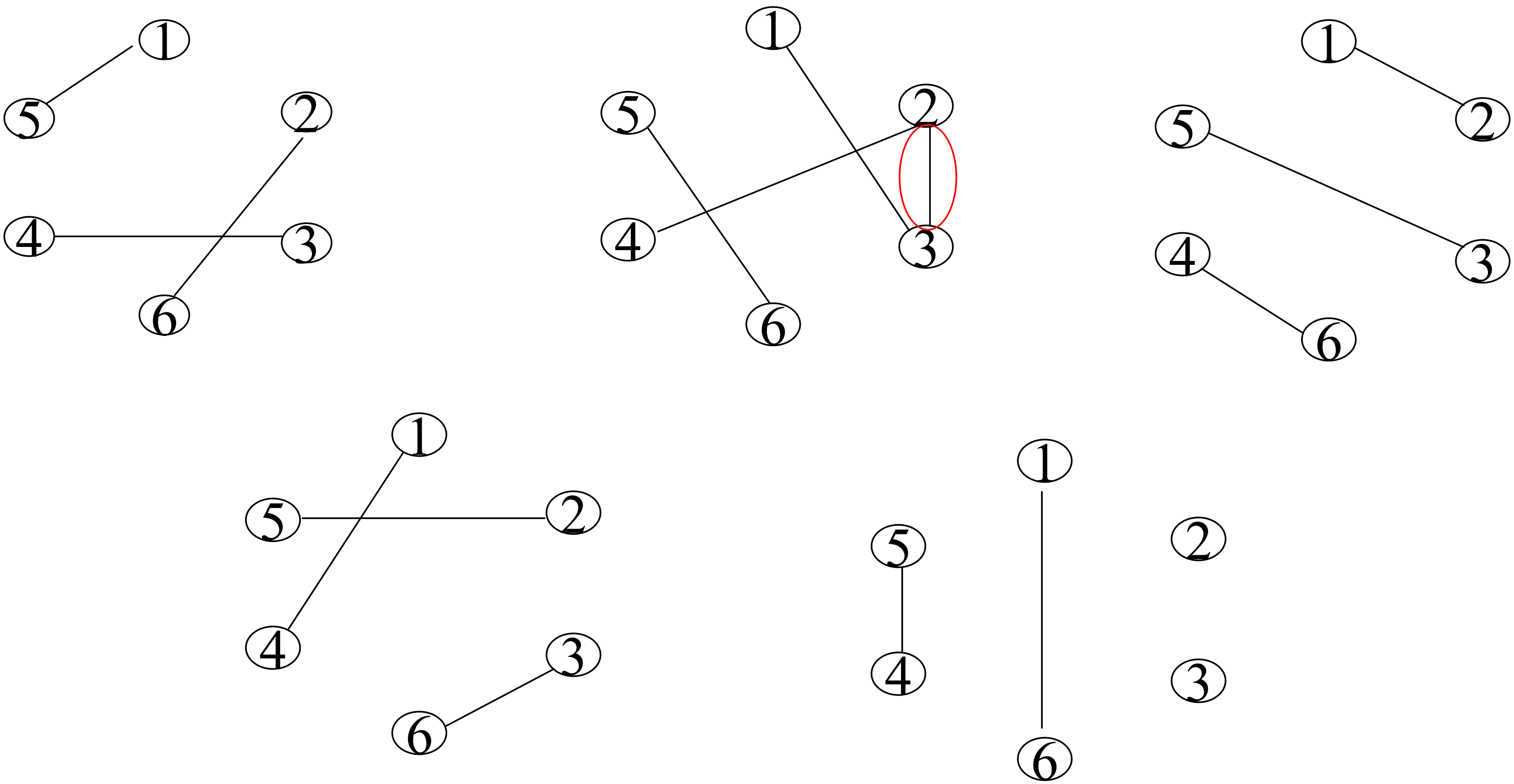
Ejection chain: game rotation



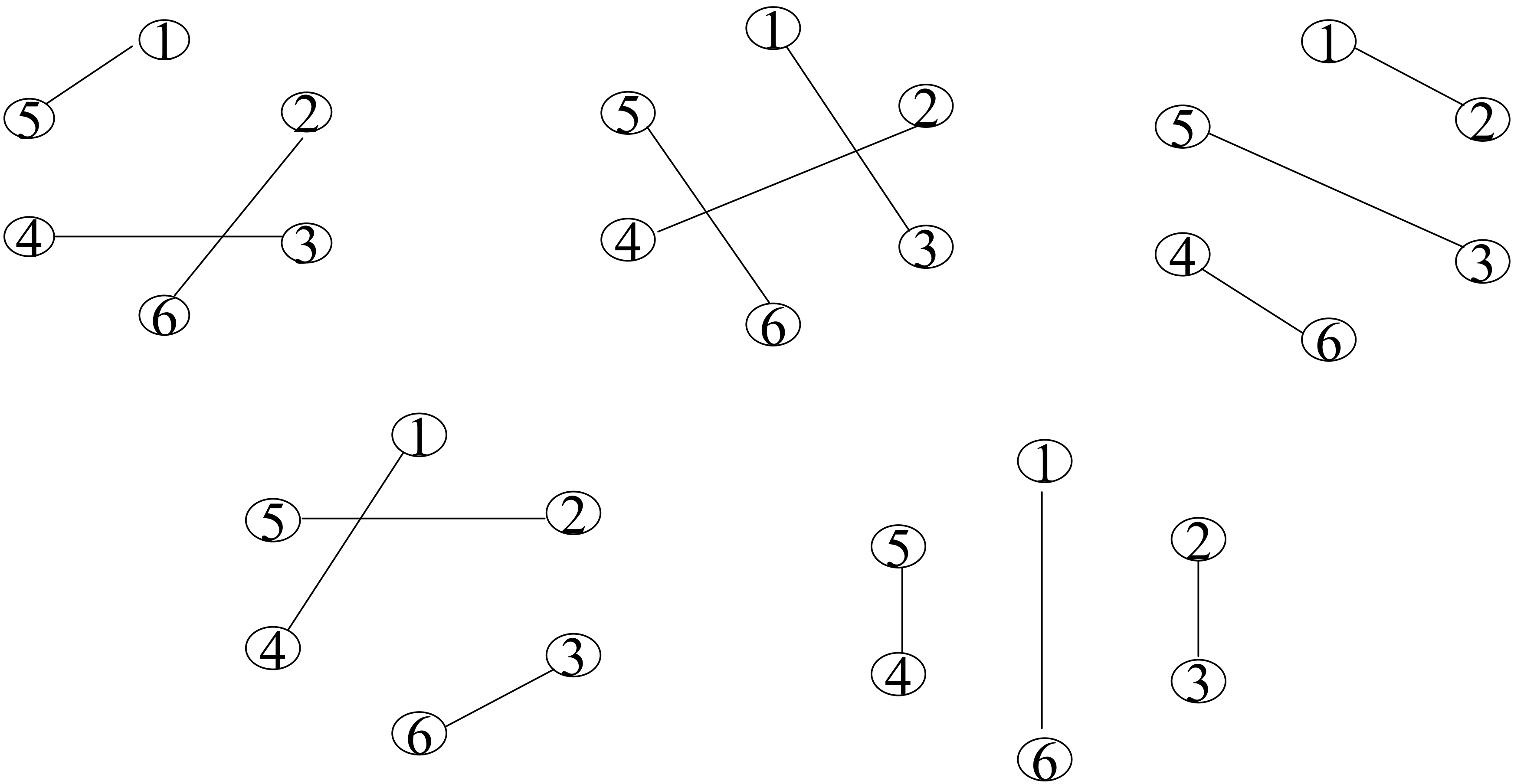
Ejection chain: game rotation



Ejection chain: game rotation



Ejection chain: game rotation



Local search

- Moves in a more complex neighborhood are explored as perturbations.
- Ejection chain:
  - Enforce a game into a given round.
  - Use an ejection chain to recover an 1-factorization.
  - Length of the ejection chain is variable.
  - Ejection chain is able to find solutions that cannot be reached by the other neighborhoods.
  - Partial round swaps may appear after an ejection chain move.
  - Only partial round swaps and game rotations may change the structure of the schedule.

GRASP + ILS heuristic

```
while .not.StoppingCriterion
    S ← GenerateRandomizedInitialSolution()
    S,S ← LocalSearch(S)        /* S best solution in cycle */
    repeat                       /* S* best overall solution */
        S' ← Perturbation(S,history)
        S' ← LocalSearch(S')
        S ← AceptanceCriterion(S,S',history)
        S* ← UpdateOverallBestSolution(S,S*)
        S ← UpdateCycleBestSolution(S,S)
    until ReinitializationCriterion
end
```

Computational results

- Circular instances with n = 12, ..., 20 teams.
- MLB instances with n = 12, ..., 16 teams.
  - All available from http://mat.gsia.cmu.edu/TOURN/
  - Largest instances exactly solved to date:
    n=6 (sequential), n=8 (20 processors in parallel on 4 days)
- Computational experiments on a Pentium IV 2.0 MHz.
- Comparisons with the best known solutions to the corresponding less constrained not necessarily mirrored instances (TTP).

Computational results

- Constructive heuristic:
  - Very fast
    - Instance MLB16: 1000 runs in approximately 1 second
  - Average gap is 17.1%.
  - Better solutions than those found after several days of computations by some metaheuristic approaches to the not necessarily mirrored version of the problem.
  - Quick constructive heuristic is very effective in terms of solution quality: other approaches start from random initial solutions.

Computational results

- GRASP + ILS heuristic: time limit is 15 minutes only.
- Before this work, times were measured in days.
- Largest gap with respect to the best known solution to the less constrained not necessarily mirrored problem was 9,5%.

Computational results

Instance	Best unmirrored	Best mirrored	gap (%)	Time to best (s)
circ12	420	456	8.6	8.5
circ14	682	714	4.7	1.1
circ16	976	1004	2.9	115.3
circ18	1420	1364	-3.9	284.2
circ20	1908	1882	-1.4	578.3
nl12	112298	120655	7.4	24.0
nl14	190056	208086	9.5	69.9
nl16	267194	285614	6.9	514.2

Computational results

Instance	Best unmirrored	Best mirrored	gap (%)	Time to best (s)
circ12	420	456	8.6	8.5
circ14	682	714	4.7	1.1
circ16	976	1004	2.9	115.3
circ18	1420	1364	-3.9	284.2
circ20	1908	1882	-1.4	578.3
nl12	112298	120655	7.4	24.0
nl14	190056	208086	9.5	69.9
nl16	267194	285614	6.9	514.2

Computational results

Instance	Best unmirrored	Best mirrored	gap (%)	Time to best (s)
circ12	420	456	8.6	8.5
circ14	682	714	4.7	1.1
circ16	976	1004	2.9	115.3
circ18	1364	1364	0.0	284.2
circ20	1882	1882	0.0	578.3
nl12	112298	120655	7.4	24.0
nl14	190056	208086	9.5	69.9
nl16	267194	285614	6.9	514.2

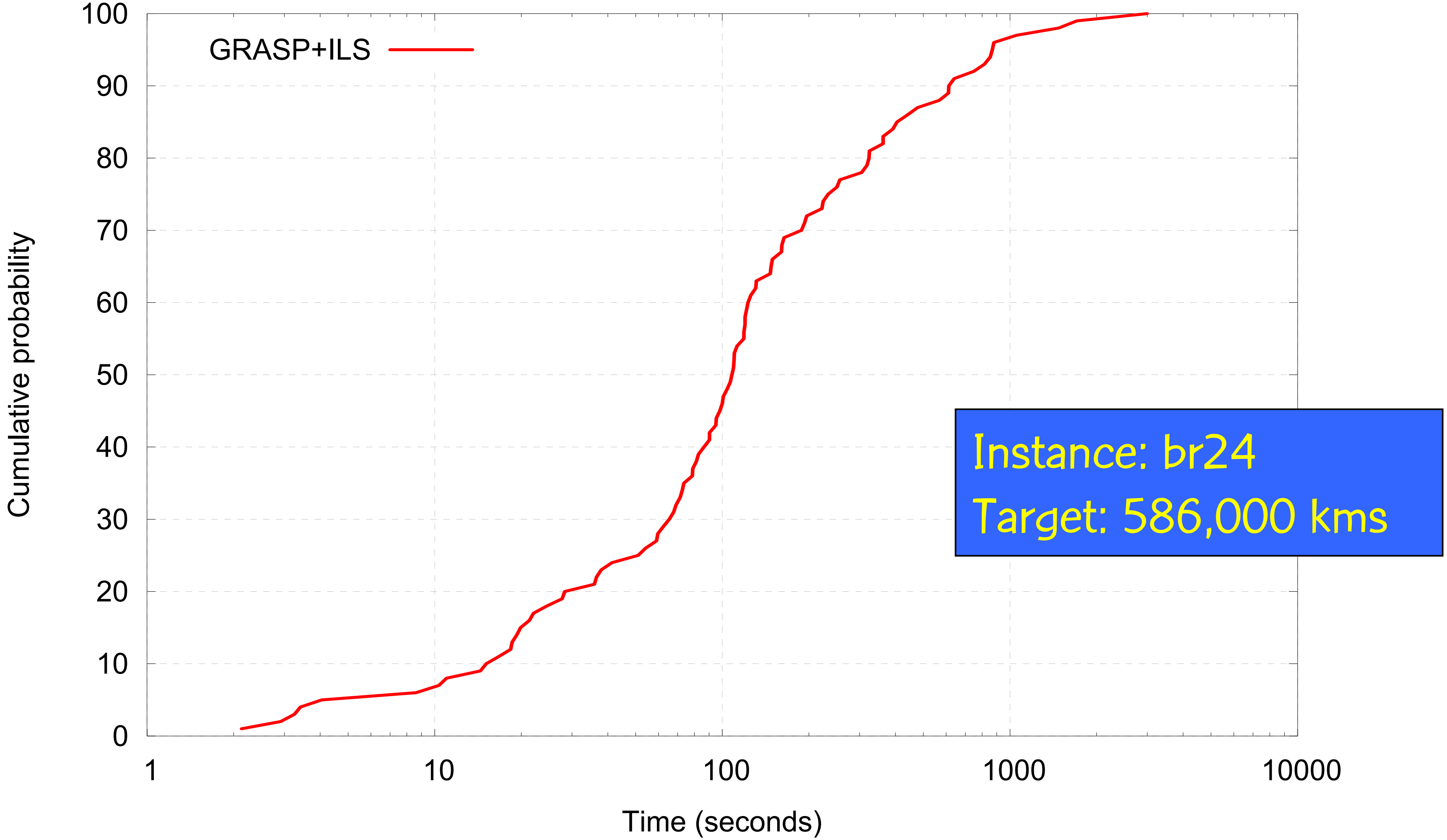
Computational results

- Total distance traveled for the 2003 edition of the Brazilian soccer championship with 24 teams (instance br24) in 12 hours (Pentium IV 2.0 MHz):

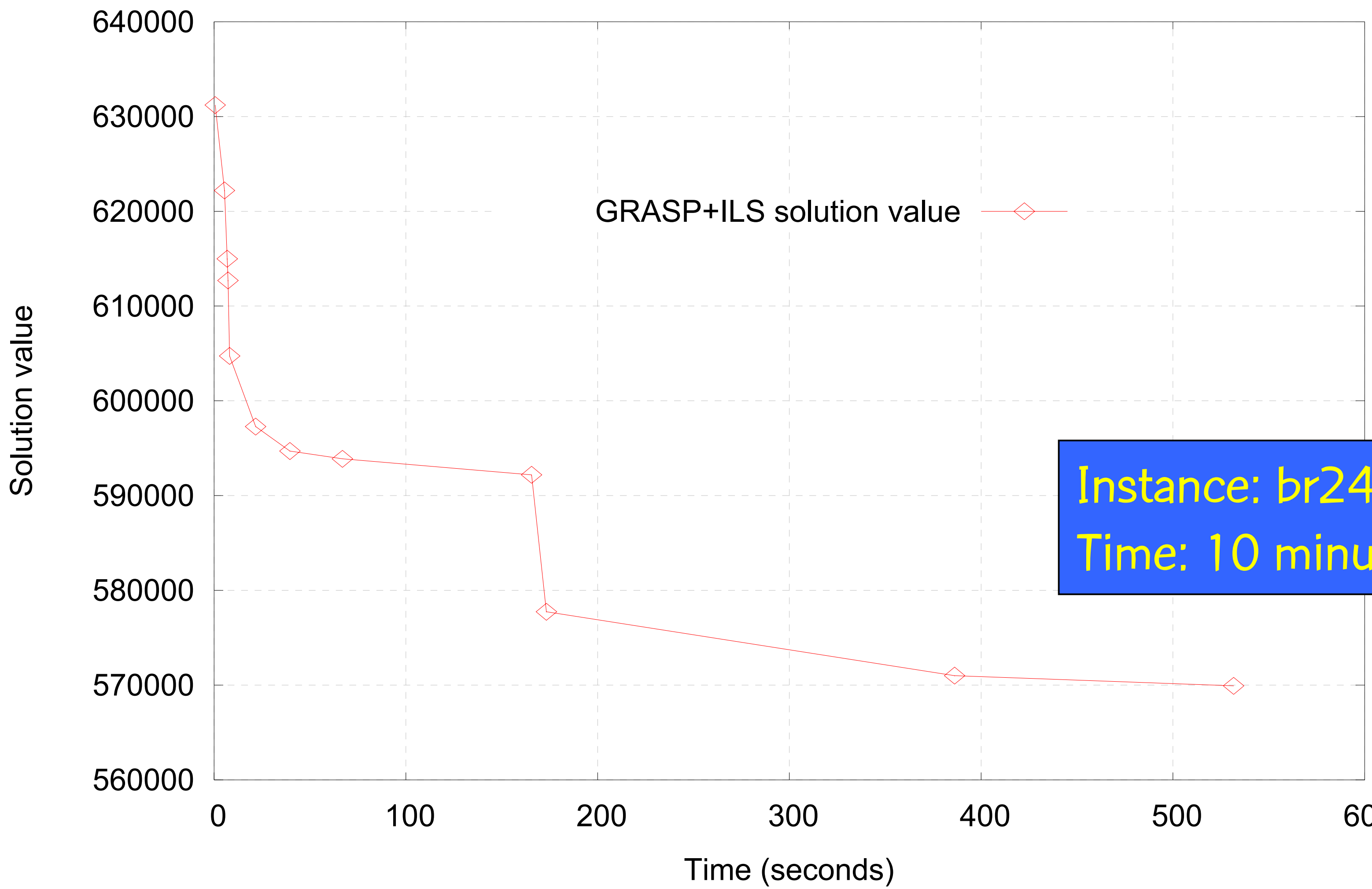
  Realized (official draw): 1, 048,134 kms

  Our solution: 506,433 kms (reduction of 52%)

Time-to-target-solution-value plot

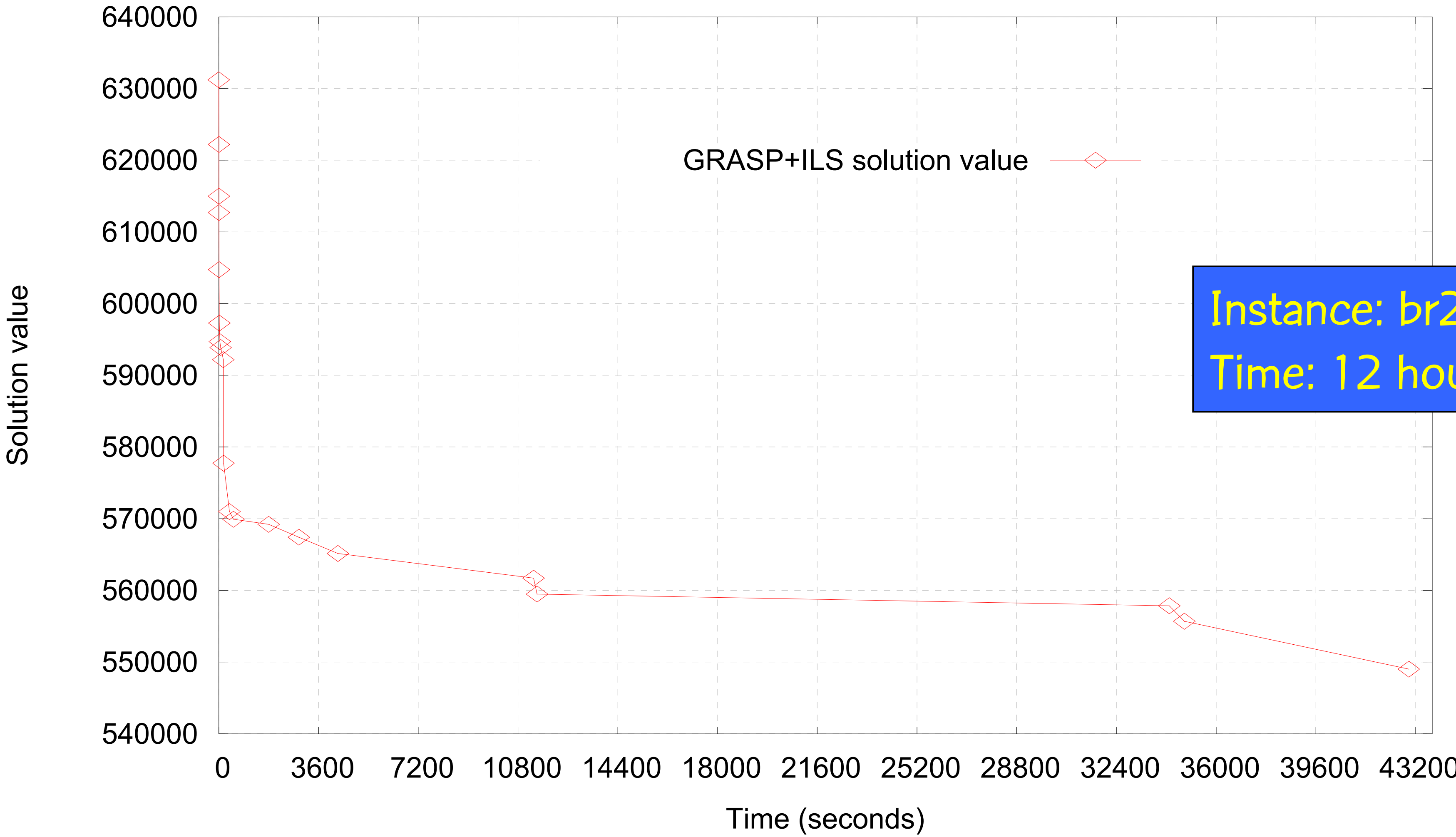


Computational results



Instance: br24
Time: 10 minutes

Computational results

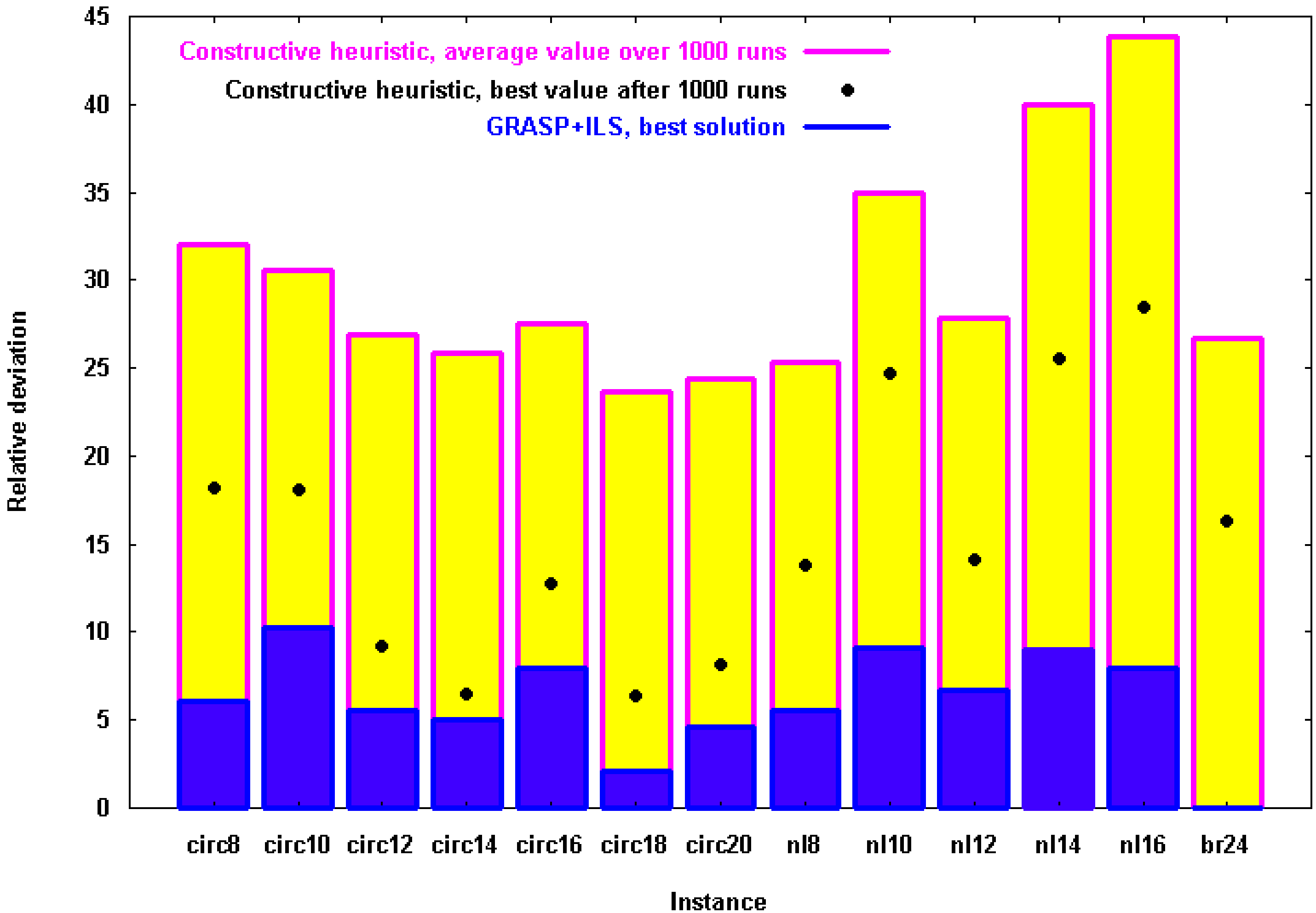


Instance: br24
Time: 12 hours

Computational results

- New heuristic improved by 3.9% and 1.4% the best known solutions to the corresponding less constrained unmirrored instances circ18 and circ20.
- Computation times are smaller than those of other heuristics, e.g. for instance MLB14:
  - Anagnostopoulos et al. (2003): approximately five days
  - GRASP + ILS: 15 minutes
- Effectiveness of the ejection chain mechanism
- Optimal solutions for a new class of constant instances (up to n=16)

Computational results



Publications

- Ribeiro & Urrutia, "Heuristics for the mirrored traveling tournament problem", *European Journal of Operational Research*, to appear.
- Ribeiro & Urrutia, "Minimizing travels by maximizing breaks in round robin tournament schedules", *Electronic Notes in Discrete Mathematics*, 2005.
- Urrutia & Ribeiro, "Maximizing breaks and bounding solutions to the mirrored traveling tournament problem", *Discrete Applied Mathematics*, to appear.

Parallel implementations

- Four different parallel versions of the GRASP + ILS sequential heuristic:
  - Parallel independent (PAR-I)
  - Parallel one-off cooperation (PAR-0) (very weak cooperation)
  - Parallel with one elite solution (PAR-1P)
  - Parallel with M elite solutions (PAR-MP) (strong cooperation)
- PAR-1P and PAR-MP are truly cooperative parallel versions.
- PAR-MP: master handles a pool of at most M elite solutions that are collected from and distributed upon request to the slaves at the end of their ILS phases.

Parallel implementations

- Algorithms implemented using:
  - C++
  - MPI-LAM (version 7.0.6)
  - Globus
  - EasyGrid middleware

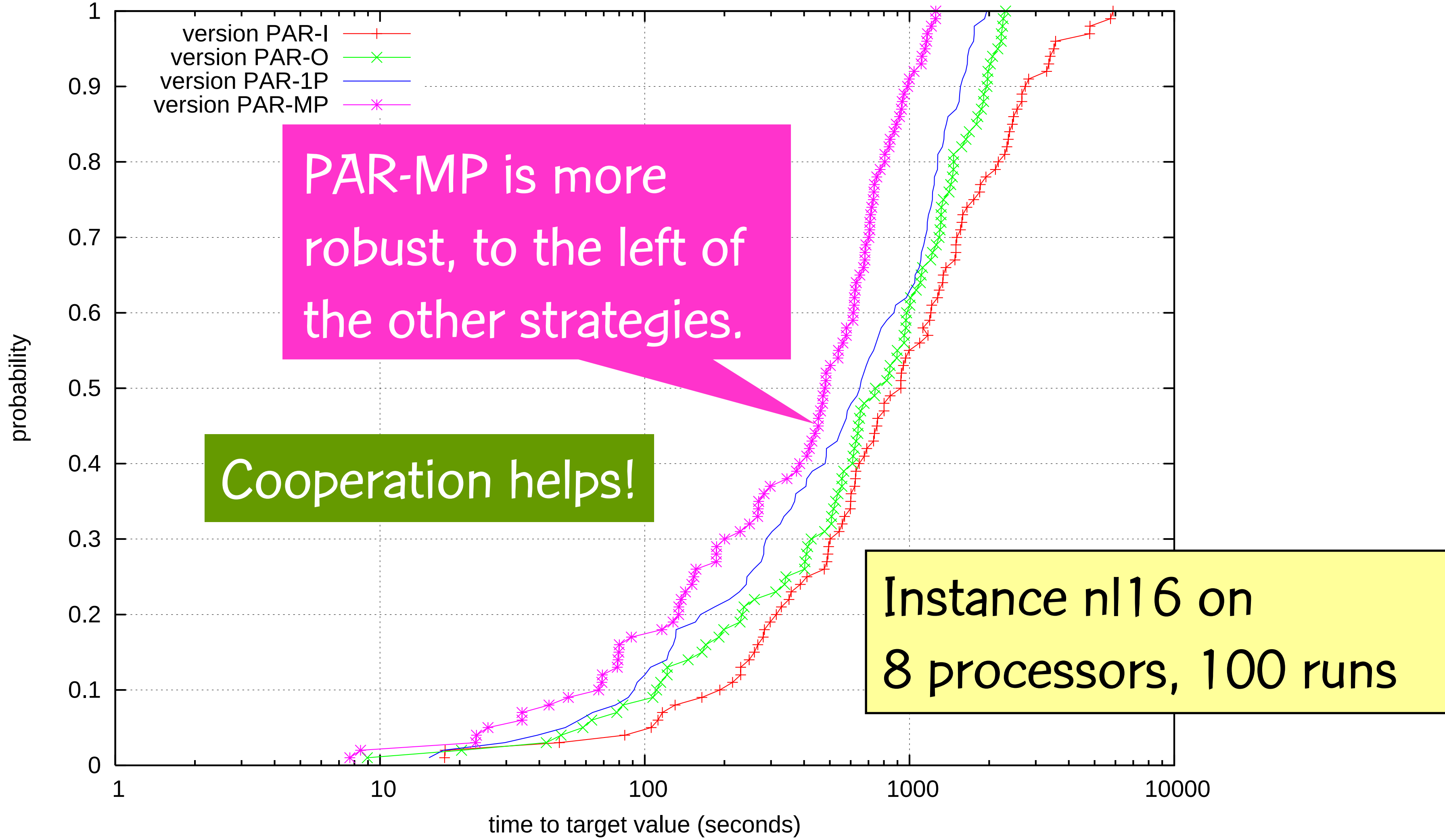
- Experiments:
  - Dedicated Linux cluster of 1.7 GHz Pentium IV machines, each with 256 Mbytes of RAM

Computational results

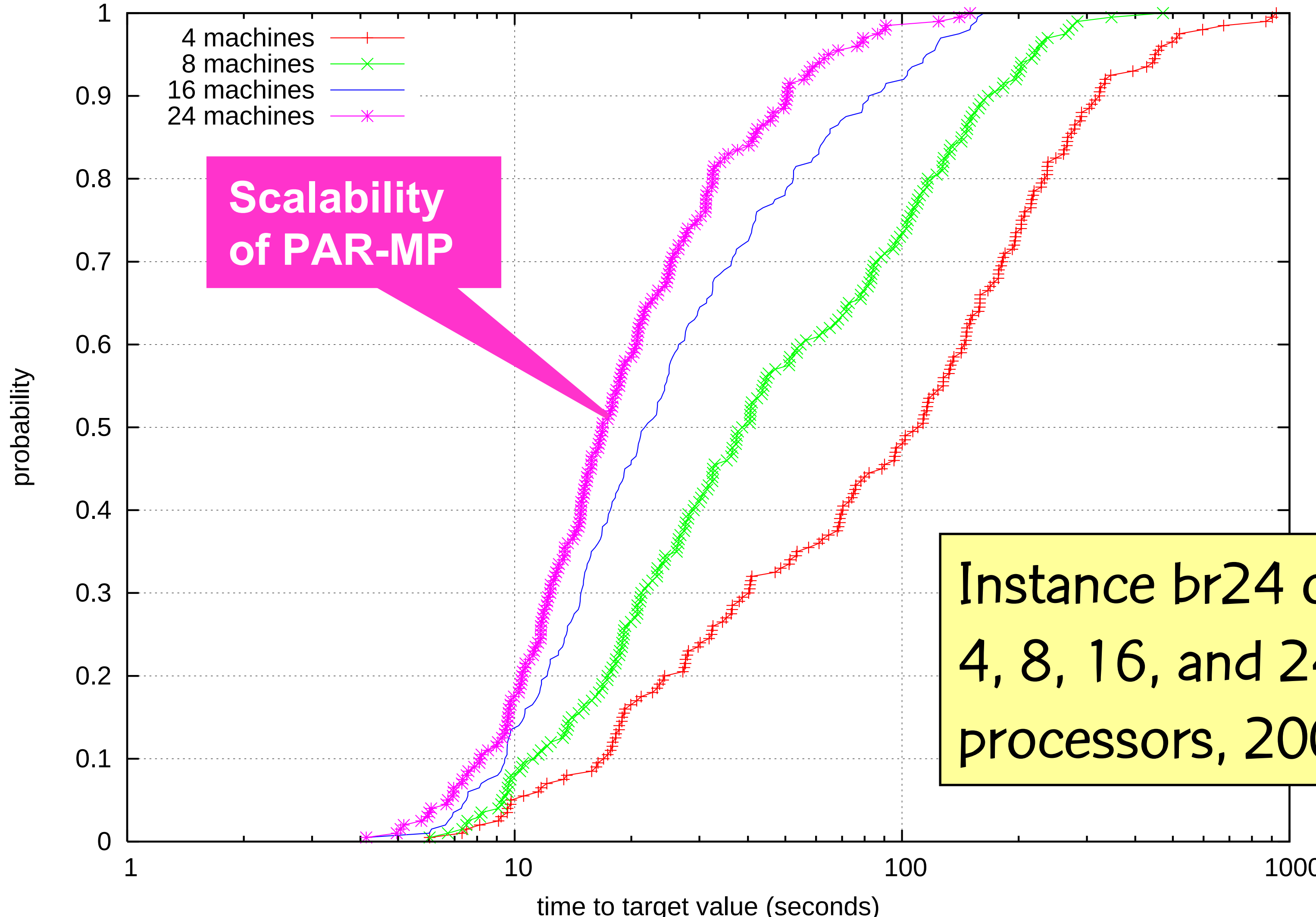
Instance	Sequential	PAR-I/O/1P	PAR-MP
circ10	276	272	272
circ16	1004	984	980
circ18	1364	1308	1306
nl16	285614	280174	279618
br24	506433	503158	503158

Better solutions found by each parallel version

Time-to-target-solution-value plot



Time-to-target-solution-value plot



Instance br24 on 4, 8, 16, and 24 processors, 200 runs

Publications

- Best known solution for TTP and MTTP instance circ18 was obtained by the PAR-MP implementation of the GRASP + ILS heuristic.
- True (shared) grid implementation running on 80 machines distributed over 3 cities (100 kms away): significant speedups for same solution quality.
- Araújo, Boeres, Rebello, Ribeiro, & Urrutia, "A grid implementation of a GRASP-ILS heuristic for the MTTP", *Metaheuristics International Conference*, 2005.

Applications in sports scheduling and management

Scheduling the Brazilian soccer championship

Brazilian soccer championship

- TTP may be the appropriate model for some US tournaments (NHL, MLB, NBA) ...
- ... but not for Brazilian tournaments!
  - MLB: a team may have to play up to 140 games in 6 months.
  - Brazil: few games in the middle of the week, teams return to their home cities after each game.
- What is the real problem to be solved?

Brazilian soccer championship

- Technical criteria:
  - Minimize the sequences of home and away games: ideally, alternate home and away games (not possible for all teams).
  - A team that plays at home in the first round will play away in the last (and vice-versa).
  - No important regional games in the last four rounds.
  - Stadium availability: all games take place at the same time in the last round.
  - Security: complementary patterns for teams in the same city (to avoid clashes of fans).

Brazilian soccer championship

- TV standards are the most important:
  - TV Globo is the major sponsor.
  - Set of more important teams: "Club of 13"
  - There should be an away game of an important team of Rio de Janeiro (resp. São Paulo) against another team of "Club of 13" to be broadcast to Rio de Janeiro (resp. São Paulo) in every round, since TV does not broadcast games to the same city where they take place.
  - Number of PPV channels available for broadcasting.
  - Availability of uplink transmission units.
- Problem: minimize the number of violated criteria.

Applications in sports scheduling and management

Referee assignment

Referee assignment problem

- Regional amateur leagues in the US (baseball, basketball, soccer): hundreds of games every weekend in different divisions
  - boys, girls
  - age: 8-10, 10-12, 12-14, 14-16, 16-18
- In a single league in California there might be up to 500 soccer games in a weekend, to be refereed by 600 certified referees.
- Problem: assign referees to games.

Referee assignment problem

- Constraints:
  - Up to three referees may be necessary for each game.
  - Games require referees with different levels of certification.
  - A referee cannot be assigned to a game where he/she is a player.
  - Timetabling conflicts and traveling times.
  - Referee groups: cliques of referees that request to be assigned to the same games (relatives, car pools).
  - Number of games a referee is willing to referee.
  - Traveling constraints.

Referee assignment problem

- Multicriteria optimization problem:
  - Difference between the number of games the referee is willing to referee and the number of games to which he/she is assigned.
  - Traveling times between different games.
- Constructive + local search heuristics
- Results soon!

Final remarks

- Slides:

  http://www.inf.puc-rio.br/~celso/talks.htm

- Papers:

  http://www.inf.puc-rio.br/~celso/publicacoes.htm

- OR applications in sports:

  http://www.esportemax.org

- Please let us know about new links and materials!