Concurrent Parameterized Games

Anirban Majumdar
Joint work with Nathalie Bertrand and Patricia Bouyer

LSV, ENS Paris-Saclay, France
Inria Rennes, France

MOVEP 2020
Turn based 2-player Games

- $P_1 (\bigcirc)$ vs $P_2 (\square)$
Turn based 2-player Games

- $P_1 (\bigcirc)$ vs $P_2 (\square)$
Turn based 2-player Games

- $P_1 (\bigcirc)$ vs $P_2 (\square)$
Turn based 2-player Games

- $P_1 (\bigcirc) \text{ vs } P_2 (\Box)$
Turn based 2-player Games

- $P_1 (\bigcirc)$ vs $P_2 (\square)$
Turn based 2-player Games

- $P_1 (\bigcirc)$ vs $P_2 (\square)$
Turn based 2-player Games

- $P_1$ (○) vs $P_2$ (□)
- Win = A set of plays
Turn based 2-player Games

• $P_1 (\bigcirc)$ vs $P_2 (\Box)$
• Win = A set of plays
• Positional strategies for Reachability, Safety…
Concurrent 2-player Games

[Alfaro, Henzinger, Kupferman '07]

- Actions: $\Sigma = \{a, b\}$
- $P_1, P_2$ - choose action *simultaneously*
- Next vertex determined by the chosen actions
Concurrent 2-player Games

[Alfaro, Henzinger, Kupferman ’07]

- Actions: $\Sigma = \{a, b\}$
- $P_1, P_2$ - choose action simultaneously
- Next vertex determined by the chosen actions
- Eg: $P_1$ chooses ‘$a$’; $P_2$ chooses ‘$b$’
The game goes from $v_0$ to $v_2$
Concurrent 2-player Games

[Alfaro, Henzinger, Kupferman ’07]

- Actions: $\Sigma = \{a, b\}$
- $P_1, P_2$ - choose action simultaneously
- Next vertex determined by the chosen actions
- Eg: $P_1$ chooses ‘$a$’; $P_2$ chooses ‘$b$’
  The game goes from $v_0$ to $v_2$
Concurrent 2-player Games

[Alfaro, Henzinger, Kupferman ’07]

- choose action simultaneously
- Actions: \(\Sigma = \{a, b\}\)
- \(P_1, P_2\) - choose action simultaneously
- Next vertex determined by the chosen actions
- Eg: \(P_1\) chooses ‘\(a\)’; \(P_2\) chooses ‘\(b\)’
  
The game goes from \(v_0\) to \(v_2\)
Concurrent k-player Games

- Actions: $\Sigma = \{a, b\}$
- $P_1, \ldots, P_k$ - choose action simultaneously
Concurrent k-player Games

- Actions: $\Sigma = \{a, b\}$
- $P_1, \ldots, P_k$ - choose action simultaneously
- Eg: $P_1$ chooses 'a'; $P_2, \ldots, P_k$ all choose 'b'

The game goes from $v_0$ to $v_2$
Concurrent parameterized Games

- Actions: $\Sigma = \{a, b\}$
- $P_1$ vs Rest of world (Env.)
- Unknown (parameter) but fixed number of players
Concurrent parameterized Games

- Actions: $\Sigma = \{a, b\}$
- $P_1$ vs Rest of world (Env.)
- Unknown (parameter) but fixed number of players

- $L_1, L_2, L_3$ Regular
- $P_1$ needs to win against all choices of others
How to play Parameterized Games
1. Env chooses $k$: #players (unknown to $P_1$)
1. Env chooses $k$: #players (unknown to $P_1$)
2. $P_1$ chooses an action ‘$a_1$’
   Others choose actions ‘$a_2, \ldots, a_k$’
   This forms a word $a_1a_2\ldots a_k$
1. Env chooses $k$: #players (unknown to $P_1$)
2. $P_1$ chooses an action ‘$a_1$’
   Others choose actions ‘$a_2, \ldots, a_k$’
   This forms a word $a_1 a_2 \ldots a_k$
3. Env chooses $v_i$ s.t. $a_1 a_2 \ldots a_k \in L_i$
How to play Parameterized Games

1. Env chooses k: #players (unknown to $P_1$)
2. $P_1$ chooses an action ‘$a_1$’
   Others choose actions ‘$a_2, \ldots, a_k$’
   This forms a word $a_1a_2\ldots a_k$
3. Env chooses $v_i$ s.t. $a_1a_2\ldots a_k \in L_i$
4. Game proceeds to $v_i$ and goto step 2
1. Env chooses \( k \): #players (unknown to \( P_1 \))

2. \( P_1 \) chooses an action ‘\( a_1 \)’
   Others choose actions ‘\( a_2, \ldots, a_k \)’
   This forms a word \( a_1a_2\ldots a_k \)

3. Env chooses \( v_i \) s.t. \( a_1a_2\ldots a_k \in L_i \)

4. Game proceeds to \( v_i \) and goto step 2

\( P_1 \) has to win against all choices of others, for all \( k \)
• **Question:** Do positional strategies suffice?

![Illustrative example diagram]

• **Objective of** $P_1$: Reach $\bigcirc$
Illustrative example

- **Question**: Do positional strategies suffice?  ❌

- Objective of $P_1$: Reach ⬤
- $P_1$ has winning strategy
- No positional winning strategy
Problem Simplification

- **Goal:** Solve Parameterized game for $P_1$

- **Observation:** Only number of opponents matter (not their choices)
Problem Simplification

- **Goal**: Solve Parameterized game for $P_1$

- **Observation**: Only number of opponents matter (not their choices)
Problem Simplification

- **Goal**: Solve Parameterized game for $P_1$

- **Observation**: Only number of opponents matter for general case also
Problem Simplification

- **Goal**: Solve Parameterized game for $P_1$

- **Observation**: Only number of opponents matter for general case also
Problem Simplification

- **Goal**: Solve Parameterized game for $P_1$

- **Observation**: Only number of opponents matter for general case also
  - $L$ regular $\Rightarrow$ set of lengths of words ($|L|$) is semilinear
Problem Simplification

- **Goal**: Solve Parameterized game for $P_1$

- **Observation**: Only number of opponents matter for general case also

- L regular $\Rightarrow$ set of lengths of words ($|L|$) is semilinear

- Different cases on the representation of $|L|$
  - Intervals
  - Unions of intervals
  - Semilinear sets
How to play the simplified game
How to play the simplified game

1. Env chooses $k$: #opponents (unknown to $P_1$)
How to play the simplified game

1. Env chooses $k$: #opponents (unknown to $P_1$)

2. $P_1$ chooses an action ‘$\sigma$’
How to play the simplified game

1. Env chooses $k$: #opponents (unknown to $P_1$)
2. $P_1$ chooses an action ‘$\sigma$’
3. Env chooses $v_i$ s.t. $k \in S_i^\sigma$
How to play the simplified game

1. Env chooses k: # opponents (unknown to $P_1$)
2. $P_1$ chooses an action ‘$\sigma$’
3. Env chooses $v_i$ s.t. $k \in S_i^\sigma$
4. Game proceeds to $v_i$ and goto step 2
How to play the simplified game

1. Env chooses k: #opponents (unknown to $P_1$)
2. $P_1$ chooses an action ' $\sigma$'
3. Env chooses $v_i$ s.t. $k \in S_i^\sigma$
4. Game proceeds to $v_i$ and goto step 2

$P_1$ has to win for all k
Resolution of the game

- Construction of Knowledge game ($\mathcal{K}$)

\[
\begin{align*}
\begin{array}{c}
v_1 \\
\end{array}
& \xrightarrow{a, [1]} \\
\begin{array}{c}
v_2 \\
\end{array}
\begin{array}{c}
v_3 \\
\end{array}
& \xrightarrow{a, [2, \infty)}
\end{align*}
\]
Resolution of the game

- Construction of Knowledge game ($\mathcal{K}$)
Resolution of the game

- Construction of Knowledge game ($\mathcal{K}$)
Illustrative example

\[ v_1, [1, \infty) \]

\[ v_2, [1] \]

\[ v_3, [2, \infty) \]

\[ v_4, [1] \]

\[ v_5, [2, \infty) \]

\[ v_6, [2, \infty) \]
Resolution of the game

- Construction of Knowledge game ($\mathcal{K}$)

$P_1$ has winning strategy $\iff$ $\bigcirc$ has winning strategy
Resolution of the game

- Construction of Knowledge game ($\mathcal{K}$)

- $\mathcal{K}$ is finite: only intersections
- Solving Parameterized game is decidable
Resolution of the game

- Construction of **Knowledge game** ($\mathcal{K}$)

$P_1$ has winning strategy $\iff$ $\bigcirc$ has winning strategy

- $\mathcal{K}$ is finite: only intersections
- Solving Parameterized game is **decidable**
- **Complexity?**
• **Complexity results:**

<table>
<thead>
<tr>
<th></th>
<th>Deterministic</th>
<th>Non-deterministic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intervals</td>
<td>PTIME-complete</td>
<td></td>
</tr>
<tr>
<td>Unions of intervals</td>
<td>NP-complete</td>
<td>PSPACE-complete</td>
</tr>
<tr>
<td>Semilinear sets</td>
<td></td>
<td>PSPACE-complete</td>
</tr>
</tbody>
</table>

• Complexities are -
  - in #endpoints for (unions of) intervals
  - in #semilinear_set for semilinear sets
Intervals

- Size of $\mathcal{K}$: quadratic in #endpoints
- Turn-based game solvable in Polynomial time (reachability)
  - Parameterized game solvable in Polynomial time (reachability)
Semilinear sets

- Size of $\mathcal{K}$: exponential in $\#\text{semilinear\_sets}$
- Polynomial-space algorithm (next...)
Semilinear Sets - PSPACE upper bound

- Solving Reachability

Step 1. Construct $\mathcal{H}[v, K]$ - restriction of $\mathcal{H}$
Semilinear Sets - PSPACE upper bound

- Solving Reachability

Step 1. Construct $\mathcal{H}[v, K]$ - restriction of $\mathcal{H}$
Semilinear Sets - PSPACE upper bound

- Solving Reachability

Step 1. Construct $\mathcal{H}[v, K]$ - restriction of $\mathcal{H}$

- Stop at any $K' \subsetneq K$
• Solving Reachability

Step 1. Construct \( \mathcal{H}[v, K] \) - restriction of \( \mathcal{H} \)

• Stop at any \( K' \not\subset K \)

• Polynomial size game : solvable in Polynomial time
Semilinear Sets - PSPACE upper bound

\[ \mathcal{K}[v_0, \mathbb{N}] \]

\[ \mathcal{K}[v_1, K_1] \]

\[ \mathcal{K}[v_2, K_2] \]

\[ \mathcal{K}[v_3, K_3] \]
Step 2. Apply DFS - reuse "space"

$v_0, K_0, ?$
Semilinear Sets - PSPACE upper bound

Step 2. Apply DFS - reuse "space"

\[ \mathcal{K}[v_0, K_0] \]
Step 2. Apply DFS - reuse "space"

Semilinear Sets - PSPACE upper bound

$\mathcal{K}[v_0, K_0]$

$\mathcal{K}[v_3, K_3]$

poly
Semilinear Sets - PSPACE upper bound

Step 2. Apply DFS - reuse “space”

\[ K[v_0, K] \]
\[ K[v_3, K] \]
\[ K[v_6, K] \]
Step 2. Apply DFS - reuse "space"

\[ \mathcal{K}[v_0, K_0] \]

\[ \mathcal{K}[v_3, K_3] \]

\[ \mathcal{K}[v_6, K_6] \]
Semilinear Sets - PSPACE upper bound

Step 2. Apply DFS - reuse “space"

- **tag**\((v, K) = \text{Win}; \) if \(v\) is target or \(P_1\) has a strategy to reach 'Win' in \(\mathcal{K}[v, K]\)

- tag once computed, the subtree is "forgotten"
Step 2. Apply DFS - reuse “space"

- tag(v,K) = Win; if either, v is target
- or, P₁ has a strategy to reach 'Win' in $\mathcal{K}[v, K]$

- tag once computed, the subtree is "forgotten"
Semilinear Sets - PSPACE upper bound

Step 2. Apply DFS - reuse "space"

- \( \mathcal{H}[v_0, K_0] \)
- \( \mathcal{H}[v_3, K_3] \)

- \( \text{tag}(v, K) = \text{Win}; \text{ if } \{ \text{either, } v \text{ is target} \)  
  \text{or, } P_1 \text{ has a strategy to reach 'Win' in } \mathcal{H}[v, K] \)

- \( \text{tag once computed, the subtree is "forgotten"} \)
Semilinear Sets - PSPACE upper bound

Step 2. Apply DFS - reuse "space"

- \( \text{tag}(v, K) = \text{Win}; \) if \( v \) is target
  - or, \( P_1 \) has a strategy to reach 'Win' in \( \mathcal{K}[v, K] \)
- \( \text{tag} \) once computed, the subtree is "forgotten"
Unions of intervals, Deterministic - NP upper bound

- Non-deterministically guess a strategy
- Size polynomial (in #endpoints)
Conclusion

Parameterized Concurrent Games

- Generalisation of 2-player concurrent games
- $P_1$ against the world
- Strategies need memory
- Knowledge game construction
- PSPACE-completeness in general case
- Better bounds for simpler cases
Future

Parameterized Synthesis

- Coalition game
- Number of players unknown
- Players know their "id"
Future

Parameterized Synthesis

- Coalition game
- Number of players unknown
- Players know their "id"
- Collective winning strategy
- Player ‘i’ plays ‘b’ at i-th round, ‘a’ otherwise
Parameterized Synthesis

- Coalition game
- Number of players unknown
- Players know their “id”
- Collective winning strategy
- Player ‘i’ plays ‘b’ at i-th round, ‘a’ otherwise

Thank You