Efficient Decomposition of Bimatrix Games

Xiang Jiang *Arno Pauly*

Computer Laboratory, University of Cambridge

SR 2014, Grenoble

Overview

We are interested in (doable) operations $\odot$ on the class of bimatrix games, such that

1. from Nash equilibria of games G_1 and G_2, we can obtain a Nash equilibrium of $G_1 \odot G_2$,
2. and from any Nash equilibrium of $G_1 \odot G_2$ we can obtain Nash equilibria of both G_1 and G_2.
3. Furthermore, $\odot^{-1}$ is doable, too.

then we may conclude:

- Solving a fixed finite number of games is no harder than solving a single game.
- Solving a large decomposable game is no harder than solving a number of small games.

Outline

The product-operator

The sum-operator

A detour on wtt-reductions

The decomposition algorithm

Defining products

Definition

Given an $n_1 \times m_1$ bimatrix game (A^1, B^1) and an $n_2 \times m_2$ bimatrix game (A^2, B^2), we define the $(n_1 n_2) \times (m_1 m_2)$ product game $(A^1, B^1) \times (A^2, B^2)$ as (A, B) with $A_{[i_1,i_2][j_1,j_2]} = A^1_{i_1 j_1} + A^2_{i_2 j_2}$ and $B_{[i_1,i_2][j_1,j_2]} = B^1_{i_1 j_1} + B^2_{i_2 j_2}$.

$$\begin{pmatrix} A^1 + A^2_{11} & A^1 + A^2_{12} & \cdots \\ A^1 + A^2_{21} & A^1 + A^2_{22} & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix}$$

Results for equilibria

Theorem

If (x^k, y^k) is a Nash equilibrium of (A^k, B^k) for both $k \in \{1, 2\}$, then (x, y) is a Nash equilibrium of (A, B), where $x_{[i_1 i_2]} = x^1_{i_1} x^2_{i_2}$ and $y_{[m_1 m_2]} = y^1_{m_1} y^2_{m_2}$.

Theorem

If (x, y) is a Nash equilibrium of (A, B), then (x^1, y^1) given by $x^1_i = \sum_{l=1}^{n_2} x_{[i,l]}$ and $y^1_j = \sum_{l=1}^{m_2} y_{[j,l]}$ is a Nash equilibrium of (A^1, B^1).

The definition

Definition

Given an $n_1 \times m_1$ bimatrix game (A^1, B^1) and an $n_2 \times m_2$ bimatrix game (A^2, B^2), we define the $(n_1 + n_2) \times (m_1 + m_2)$ sum game $(A^1, B^1) + (A^2, B^2)$ via the constant $K > \max_{i,j}\{|A_{i,j}|, B_{i,j}|\}$ as (A, B) with:

$$A_{i,j} = \begin{cases} A^1_{ij} & \text{if } i \leq n_1, j \leq m_1 \\ A^2_{(i-n_1),(j-m_1)} & \text{if } i > n_1, j > m_1 \\ K & \text{otherwise} \end{cases}$$

$$B_{i,j} = \begin{cases} B^1_{ij} & \text{if } i \leq n_1, j \leq m_1 \\ B^2_{(i-n_1),(j-m_1)} & \text{if } i > n_1, j > m_1 \\ -K & \text{otherwise} \end{cases}$$

$$\begin{pmatrix} A^1 & K \\ K & A^2 \end{pmatrix} \qquad \begin{pmatrix} B^1 & -K \\ -K & B^2 \end{pmatrix}$$

The first result on equilibria

Lemma

Let (x, y) *be a Nash equilibrium of* $(A^1, B^1) + (A^2, B^2)$*. Then* $0 < \left(\sum_{i=1}^{n_1} x_i\right) < 1$ *and* $0 < \left(\sum_{j=1}^{m_1} y_j\right) < 1$*.*

Theorem

If (x, y) *is a Nash equilibrium of* $(A^1, B^1) + (A^2, B^2)$*, then a Nash equilibrium* (x^1, y^1) *of* (A^1, B^1) *can be obtained as* $x_i^1 = \frac{x_i}{\sum_{l=1}^{n_1} x_l}$ *and* $y_j^1 = \frac{y_i}{\sum_{l=1}^{m_1} y_l}$*.*

The second result on equilibria

Theorem

Let (x^k, y^k) be a Nash equilibrium of (A^k, B^k) resulting in payoffs (P^k, Q^k) for both $k \in \{1, 2\}$. Then (x, y) is a Nash equilibrium of $(A^1, B^1) + (A^2, B^2)$, where $x_i = x_i^1 \frac{K - Q^2}{2K - Q^1 - Q^2}$ for $i \leq n_1$, $x_i = x_{i-n_1}^2 \frac{K - Q^1}{2K - Q^1 - Q^2}$ for $i > n_1$, $y_j = y_j^1 \frac{K - P^2}{2K - P^1 - P^2}$ for $j \leq m_1$, $y_j = y_{j-m_1}^2 \frac{K - P^1}{2K - P^1 - P^2}$ for $j > m_1$.

Complexity of the translations

The operations on games and Nash equilibria are very simple.
In particular, they are

- computable even for real payoffs.
- BSS computable.
- polynomial-time computable for rational payoffs.

wtt and many-one reductions coincide

The product was originally introduced to show that $\mathrm{Nash} \equiv_W \mathrm{Nash}^*$, i.e. that computable many-one reductions and computable wtt reductions to the problem of finding a Nash equilibrium for a bimatrix game with real-valued payoffs coincide.
Similarly, the sum can be used to show polynomial-time many-one and polynomial-time wtt reductions to finding Nash equilibria of bimatrix games with rational payoffs coincide. This implies that PPAD is closed under polynomial-time wtt reductions.

The algorithm

Our basic algorithm proceeds as follows: To solve a game (A, B)

1. test whether (A, B) is the sum of (A^1, B^1) and (A^2, B^2) via some constant K. If yes, solve (A^1, B^1) and (A^2, B^2) and combine their Nash equilibria to an equilibrium of (A, B) via Theorem 7. If no,
2. test whether (A, B) is the product of (A^1, B^1) and (A^2, B^2). If yes, solve (A^1, B^1) and (A^2, B^2) and combine their Nash equilibria to an equilibrium of (A, B) via Theorem 2. If no,
3. find a Nash equilibrium of (A, B) by some other means (i.e. the GAMBIT library).

Runtime: $O(S^2 f(\lambda))$

Experimental procedure

1. We randomly generated a *decomposition tree*
2. and payoffs in the leaves of size $\leq 3 \times 3$,
3. such that the entire game would have size $(100 - -110) \times (100 - -1110)$.
4. Then we forgot about the structure, and only consider the resulting bimatrix game.

Some experimental results

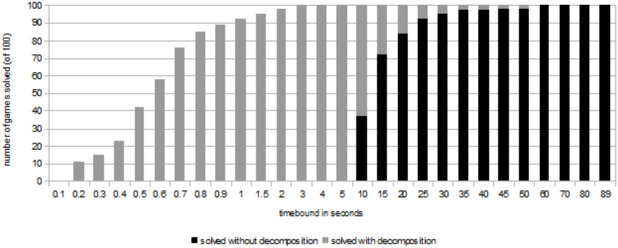

Figure: gambit-gnm

Questions?

Thanks for listening.