

POLITECNICO DI MILANO  
Scuola di Ingegneria Industriale e dell'Informazione  
Corso di Laurea Magistrale in Ingegneria Informatica  
Dipartimento di Elettronica, Informazione e Bioingegneria



Sulla computazione della Curva di Pareto  
nei giochi bimatrice

Relatore: Prof. Nicola Gatti

Tesi di laurea di:  
Andrea Panella Matr. 820177

Anno Accademico 2014-2015

*«Considerate la vostra semenza:  
fatti non foste a viver come bruti,  
ma per seguir virtute e canoscenza»*

Dante Alighieri

# Ringraziamenti

Desidero ringraziare tutte le persone che mi hanno supportato (e sopportato) durante lo svolgimento di questa tesi e più in generale durante questi cinque anni di università: senza di esse non sarei arrivato a questo traguardo. Il primo ringraziamento è rivolto al Prof. Nicola Gatti per l'aiuto, la competenza e la disponibilità dimostrate nelle vesti di relatore per questa tesi. Ringrazio il Prof. Roberto Lucchetti per avermi saputo trasmettere la passione per questa splendida disciplina chiamata Teoria dei giochi. Un grande grazie a mio padre Nicola per la pazienza biblica che dimostra ogni giorno nel sopportarmi. Grazie a mia madre Emanuela per essermi stata vicina nei momenti in cui ne avevo bisogno. Grazie a mia sorella Elisa per l'affetto che mi dimostra sempre. Grazie al Prof. Livio Bernardini per essere stato un modello di vita a trecentosessanta gradi, e per avermi ispirato nella scelta di una facoltà ingegneristica.

Infine, il grazie più grande è per i miei nonni che in tutti questi anni sono stati per me dei secondi genitori, senza se e senza ma. Questo e molti altri traguardi, sono stati possibili grazie a voi. Spero siate orgogliosi.

# Sommario

La proprietà di Pareto efficienza caratterizza soluzioni per problemi di ottimizzazione multi-obiettivo. Ricercare soluzioni con questa proprietà è nel caso generale computazionalmente intrattabile, anche ricercando soluzioni approssimate. La maggior parte dei risultati in letteratura forniscono differenti metodologie per ricercare soluzioni approssimate. La proprietà di Pareto efficienza è centrale anche nel campo della teoria dei giochi, in quanto è richiesta in svariati concetti di soluzione sia nella teoria cooperativa che non cooperativa. Non sono noti risultati di complessità computazionale per la ricerca della Curva di Pareto (tutti i punti pareto efficienti) per giochi non cooperativi. Questo documento ha l'obiettivo di fornire risultati per giochi in forma normale a 2 giocatori. Inizialmente viene fornita una classificazione dei giochi bimatrice  $2 \times 2$  rispetto alle caratteristiche geometriche dell'insieme dei possibili outcome del gioco. Successivamente si deriva un catalogo esaustivo della forma che la Curva di Pareto può assumere per questi giochi, e si propone un algoritmo in grado di derivarla in modo esatto. Infine, utilizzando un approccio *divide et impera*, si estende il risultato al caso di giochi bimatrice con un numero arbitrario di mosse, dimostrando che è possibile calcolare esattamente la Curva di Pareto in un tempo polinomiale rispetto al numero di mosse dei giocatori, e che questa è formata da un numero di pezzi che cresce, al più, polinomialmente rispetto al numero delle mosse.

# Abstract

Pareto efficiency property characterizes solutions for multi-objective optimization problems. Searching for solutions with this property is, in the general case, computationally intractable, even in the case of approximate solutions. Most of the results in literature provide different methodologies to find approximate solutions. Pareto efficiency plays a central role even in the Game of Theory area, because it is a required property for many solution concepts both in cooperative and non-cooperative theory. To the best of our knowledge, there are no computational complexity results in literature on the problem of searching the Pareto Curve (all of the Pareto efficient points) for non-cooperative games. This document aims to provide results for 2-players normal-form games. First we provide a classification of  $2 \times 2$  bimatrix games with respect to the geometrical characteristics of the feasible outcomes set of the game. Then we derive an exhaustive catalog of the possible Pareto Curve shape in this game, and we provide an algorithm to calculate it exactly. Finally, using a *divide et impera* approach, we extend the previous result to generic bimatrix games, with an arbitrary number of actions per-player, showing that it is possible to compute exactly the Pareto Curve in a polynomial time in the number of actions of the players, and that it is formed by a number of pieces that grows at most polynomially in the number of actions.

# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Preliminari e stato dell'arte</b>	<b>6</b>
1.1 Modelli di gioco . . . . .	6
1.1.1 Assunzioni . . . . .	7
1.1.2 Giochi in forma strategica . . . . .	8
1.1.3 Giochi in forma estesa . . . . .	11
1.2 Concetti di soluzione . . . . .	11
1.2.1 Equilibrio di Nash . . . . .	13
1.2.2 Pareto Efficienza . . . . .	14
1.2.3 Relazioni . . . . .	15
1.3 Complessità computazionale dei solution concepts . . . . .	18
1.3.1 Cenni di complessità computazionale . . . . .	18
1.3.2 Complessità della ricerca di un equilibrio di Nash . . . . .	21
1.3.3 Complessità soluzioni Pareto efficienti . . . . .	23
<b>2 La curva di Pareto</b>	
<b>nei giochi <math>2 \times 2</math></b>	<b>25</b>
2.1 L'insieme $Imm(\mathbf{u})$ . . . . .	25
2.1.1 Preambolo . . . . .	25
2.1.2 Involuppo di una famiglia di curve . . . . .	26
2.1.3 Proprietà della frontiera $\partial Imm(\mathbf{u})$ . . . . .	33
2.1.4 Classificazione di $Imm(\mathbf{u})$ . . . . .	35
2.1.5 Computazione di $\partial Imm(\mathbf{u})$ . . . . .	44
2.2 La Curva di Pareto . . . . .	48
2.2.1 1 vettore di payoffs puri Pareto efficiente . . . . .	49
2.2.2 2 vettori di payoffs puri Pareto efficienti . . . . .	49
2.2.3 3 vettori di payoffs puri Pareto efficienti . . . . .	59
2.2.4 4 vettori di payoffs puri Pareto efficienti . . . . .	66
2.2.5 Considerazioni . . . . .	72

<b>3 La curva di Pareto</b>	
<b>nei giochi <math>n \times n</math></b>	<b>76</b>
3.1 Divide et impera . . . . .	76
3.2 Complessità temporale . . . . .	81
3.3 Analisi sperimentale . . . . .	83
3.4 Algoritmi per concetti di soluzione basati sulla Curva di Pareto . . .	92
<b>Conclusioni e sviluppi futuri</b>	<b>93</b>
<b>Bibliografia</b>	<b>97</b>

# Elenco delle figure

1.1	Dilemma del prigioniero: Curva di Pareto e NE. . . . .	16
1.2	Sasso, carta, forbici: Curva di Pareto e NE. . . . .	16
1.3	La battaglia dei sessi : Curva di Pareto e NE. . . . .	17
1.4	La serata di gala : Curva di Pareto e NE. . . . .	18
1.5	Diverse relazioni tra le classi di complessità. . . . .	21
2.1	Albero di classificazione di $Imm(\mathbf{u})$ . . . . .	35
2.2	$Imm(\mathbf{u})$ Quadrilatero convesso. . . . .	36
2.3	$Imm(\mathbf{u})$ quadrilatero non convesso. . . . .	37
2.4	$Imm(\mathbf{u})$ Non quadrilatero. . . . .	40
2.5	$Imm(\mathbf{u})$ Non quadrilatero lineare a tratti. . . . .	41
2.6	$Imm(\mathbf{u})$ non quadrilatero degenerare 1. . . . .	42
2.7	$Imm(\mathbf{u})$ non quadrilatero degenerare 2. . . . .	42
2.8	PC 1 vettore di payoffs puri. . . . .	48
2.9	PC 2 vettori di payoffs puri: segmento. . . . .	49
2.10	PC 2 vettori di payoffs puri: Parabola. . . . .	51
2.11	PC 2 vettori di payoffs puri: 2 segmenti. . . . .	51
2.12	PC 2 vettori di payoffs puri: 2 segmenti, parabola. . . . .	52
2.13	PC 2 vettori di payoffs puri: segmento, parabola. . . . .	53
2.14	PC 2 vettori di payoffs puri: segmento, 1 discontinuità. . . . .	55
2.15	PC 2 vettori di payoffs puri: segmento, parabola, 1 discontinuità. . . . .	55
2.16	PC 2 vettori di payoffs puri: parabola, discontinuità. . . . .	56
2.17	PC 2 vettori di payoffs puri: parabola, 2 discontinuità. . . . .	57
2.18	PC 2 vettori di payoffs puri: punti isolati. . . . .	58
2.19	PC 3 vettori di payoffs puri: segmento. . . . .	59
2.20	PC 3 vettori di payoffs puri: 2 segmenti. . . . .	60
2.21	PC 3 vettori di payoffs puri: 2 segmenti, parabola. . . . .	61
2.22	PC 3 vettori di payoffs puri: 3 segmenti. . . . .	62
2.23	PC 3 vettori di payoffs puri: 3 segmenti, parabola. . . . .	62
2.24	P.C. 3 payoffs puri: 2 segmenti, 1 discontinuità . . . . .	63
2.25	PC 3 vettori di payoffs puri: segmento, parabola, 1 discontinuità. . . . .	64

2.26	PC 3 vettori di payoffs puri: 2 segmenti, parabola, 1 discontinuità. . . . .	65
2.27	PC 4 vettori di payoffs puri: segmento. . . . .	66
2.28	P.C. 4 payoffs puri: 2 segmenti . . . . .	67
2.29	PC 4 vettori di payoffs puri: 3 segmenti. . . . .	68
2.30	PC 4 vettori di payoffs puri: 4 segmenti. . . . .	68
2.31	PC 4 vettori di payoffs puri: 3 segmenti, 2 discontinuità. . . . .	69
2.32	PC 4 vettori di payoffs puri: 3 segmenti, parabola, 2 discontinuità. . . . .	70
2.33	P.C. 4 payoffs puri: 2 segmenti, parabola, 1 discontinuità . . . . .	71
2.34	PC 4 vettori di payoffs puri: 2 segmenti, 1 discontinuità. . . . .	71
3.1	Curva di Pareto della bimatrice $5 \times 5$ in Tabella 3.2. . . . .	85
3.2	Curva di Pareto della bimatrice $5 \times 5$ in Tabella 3.3. . . . .	87
3.3	Curva di Pareto della bimatrice $10 \times 10$ in Tabella 3.4. . . . .	89
3.4	Curva di Pareto della bimatrice $10 \times 10$ in Tabella 3.5. . . . .	91

# Elenco delle tabelle

1.1	Bimatrice di Sasso, carta, forbici. . . . .	9
1.2	Bimatrice del dilemma del prigioniero . . . . .	9
1.3	Bimatrice della battaglia dei sessi. . . . .	10
1.4	Bimatrice del dilemma del prigioniero: eliminazione strategia dominata	12
1.5	Bimatrice del dilemma del prigioniero: outcome . . . . .	12
1.6	Bimatrice della serata di gala. . . . .	18
2.1	Classificazione dettagliata $Imm(\mathbf{u})$ . . . . .	43
2.2	Numero di PC di una bimatrice $2 \times 2$ . . . . .	72
2.3	Riepilogo Curve di Pareto $2 \times 2$ (1) . . . . .	73
2.4	Riepilogo Curve di Pareto $2 \times 2$ (2) . . . . .	74
2.5	Riepilogo Curve di Pareto $2 \times 2$ (3) . . . . .	75
3.1	Casistica sovrapposizione intervalli di $subtract(P(U), P(V))$ . . . . .	80
3.2	Bimatrice casuale $5 \times 5$ . . . . .	84
3.3	Bimatrice casuale $5 \times 5$ . . . . .	86
3.4	Bimatrice casuale $10 \times 10$ . . . . .	88
3.5	Bimatrice casuale $10 \times 10$ . . . . .	90

# Elenco degli algoritmi

2.1	Classificazione $Imm(\mathbf{u})$ . . . . .	46
2.2	Calcolo di $\partial Imm(\mathbf{u})$ . . . . .	47
3.1	Calcolo di $P(N)$ . . . . .	78
3.2	Struttura di $subtract(P(U), P(V))$ . . . . .	79

# Introduzione

Il campo della Teoria dei giochi, inteso come strumenti, concetti e punti di vista che questa teoria ha sviluppato, ha negli ultimi 60 anni acquisito un ruolo di crescente importanza sia nell'interesse della comunità scientifica internazionale, sia in un insieme trasversale di discipline nelle quali trova concreta applicazione. La definizione di tale disciplina come: "Interactive Decision Theory" fornita in [1] è particolarmente descrittiva di alcune delle domande principali a cui questa teoria mira a rispondere: *Quali sono le decisioni migliori da intraprendere quando siamo in un ambiente in cui inter-agiamo con altri soggetti? Che cosa vuol dire migliori? In che modo le mie decisioni influenzano il comportamento e quindi a monte, le decisioni degli altri?*

Nonostante la Teoria dei giochi (Tdg) moderna nasca come parente stretta della scienza economica [17], la naturalezza e l'ampio spettro di collocazione di queste domande ha diffuso l'utilizzo degli strumenti forniti da Tdg in un insieme estremamente eterogeneo di discipline: Biologia, Sociologia, Psicologia, Filosofia, Computer security, Informatica teorica, Logica, Intelligenza artificiale, Sicurezza fisica e molte altre.

Da un punto di vista ingegneristico, particolarmente fruttuoso è stato il matrimonio tra gli strumenti forniti dall'informatica teorica e i concetti di soluzione di un gioco, forniti dalla Tdg. È sempre più centrale infatti oggi il ruolo che hanno i calcolatori in contesti multi-agente (i.e. interattivi) in cui questi devono prendere autonomamente delle decisioni, quelle migliori (per qualche definizione di migliore). La ricerca di algoritmi efficienti per determinare le decisioni migliori da intraprendere è quindi un tema estremamente attuale e rilevante, a cui la comunità scientifica sta rivolgendo una certa attenzione. Nello specifico, il vasto utilizzo della proprietà di Pareto efficienza [28] in molti concetti di soluzione e la mancanza di risultati teorici estesi riguardo alla complessità computazionale del calcolo di tutte le soluzioni Pareto Efficienti in specifiche classi di giochi, sono la motivazione principale di questo lavoro.

## Contributi

Sebbene vi siano evidenze dei primi elementi di Teoria dei giochi già nel XVIII secolo [12] e vi siano primi risultati nel XIX [10] e nel XX: [25, 37], la nascita della teoria dei giochi moderna viene comunemente ricondotta al libro pubblicato nel 1944 da Morgenstern e Von Neumann "Theory of Games and Economic Behavior" [17]. Con questo testo vengono infatti posti gli assiomi fondamentali per gli sviluppi successivi della disciplina, tra cui: la razionalità del comportamento degli agenti (i.e. i giocatori), le definizioni di utilità e di gioco e in particolare per il filone della teoria cooperativa. Circa una decina di anni dopo, J. Nash, figura correntemente nota nella cultura popolare e recentemente scomparso, in nei lavori [21–24] ha proposto un modello non cooperativo e propone al mondo il concetto fondamentale di soluzione, conosciuto poi come «*Equilibrio di Nash*» (NE) che gli è valso il premio Nobel. Negli stessi anni L. S. Shapley fornisce importanti contributi nell'area cooperativa [30] dove definisce il cosiddetto *Core* (nucleo) di soluzioni, e l'indice di potere/soluzione «*Valore di Schapley*».

Dagli inizi dei 40' sino a metà degli anni 50', in una quindicina di anni posti circa a metà del XX secolo, viene di fatto formulato quello che sarà poi ed è tutt'ora il framework fondamentale sul quale Tdg si sviluppa e sul quale un insieme sempre più numeroso e trasversale di discipline integra le proprie ricerche. Dal nostro punto di vista sembra abbastanza ragionevole considerare quel periodo come il più prolifico dal punto di vista di risultati teorici conseguiti. Nei successivi 30 anni i raffinamenti e le estensioni delle idee sviluppate in precedenza hanno portato a risultati originali di importanza indiscutibile, tra cui menzioniamo: folk theorem per giochi ripetuti [33], subgame perfect equilibria [29], equilibri correlati [2].

Contemporaneamente sono apparsi i primi lavori che utilizzano la Tdg in contesti totalmente differenti e indipendenti [5, 7, 31, 34, 35].

Ad oggi la trasversalità delle applicazioni di questa teoria ha fatto sì che i diversi concetti di soluzione formulati in 70 anni di ricerca trovassero il loro naturale impiego a seconda del campo di utilizzo. Lo studio della complessità computazionale degli equilibri di un gioco mediante un calcolatore rappresenta quindi una dei principali obiettivi (se non l'unico) dell'area di ricerca all'intersezione tra Computer Science e Tdg, questa area è comunemente conosciuta con il nome di *Algorithmic Game Theory* (Agt).

## Obbiettivi

Diversi risultati sono noti per il calcolo e la verifica di equilibri di Nash [9, 11] che riveste un ruolo centrale in situazioni non cooperative e recentemente anche per il raffinamento di tale concetto di sol che prende il nome di Strong Nash (SNE) [15], il quale contempla la possibilità per gli agenti di formare coalizioni. Questo concetto di soluzione stringe i vincoli già imposti dall'equilibrio di Nash, richiedendo anche la proprietà di Pareto Efficienza (debole) rispetto a qualsiasi possibile coalizione. Questa proprietà formulata dal Vilfredo Pareto nel 1906 in "Manuale di economia politica" [28]. caratterizza un'allocazione di beni in cui non esiste una redistribuzione che soddisfi maggiormente almeno un individuo, senza soddisfarne di meno un altro rispetto all'allocazione originale. Un'altra definizione equivalente fa leva sul famoso concetto di *miglioramento Paretiano*: un cambiamento di stato da A a B è un miglioramento Paretiano se almeno un individuo riesce ad essere più soddisfatto di prima senza che nessun altro lo sia di meno. B è Pareto ottimo (efficiente) se non esistono miglioramenti Paretiani da esso.

Pur essendo formulata più di un secolo fa' ed assumendo quindi la veste di uno dei concetti di soluzione più vecchi, la Pareto efficienza riveste ancora oggi un ruolo centrale in contesti cooperativi e non cooperativi e, più in generale, nei problemi multi-obiettivo. Oltre a SNE, diversi concetti di soluzione risultano essere Pareto efficienti, tra cui: *Utilitarian Bargaining Solution* (UBS) [32], *Nash Bargaining Solution* (NBS) [24], *Kalai-Smorodinsky Bargaining Solution*. La *Curva di Pareto* (PC) è comunemente definita come la collezione dei punti di Pareto efficienti ed in generale (per funzioni obiettivo non convesse) il calcolo di tale curva è computazionalmente difficile [3, 13, 27].

La maggior parte della letteratura scientifica si focalizza sull'approssimare la Curva di Pareto mediante rappresentazioni concise [16, 20, 27], fornire risultati per specifici casi in cui gli obbiettivi sono convessi [14] e sullo sviluppo di approcci evolutivi per generare punti approssimativamente ParetoEfficienti [8]. In questo lavoro ci focalizziamo su giochi bimatrice, i.e. giochi non cooperativi a 2 giocatori: mentre il problema di verifica della Pareto efficienza di una soluzione è computazionalmente facile [15] ad oggi non sono noti risultati del problema di determinare e caratterizzare la Curva di Pareto.

Poichè per caratterizzare la curva di Pareto di un gioco  $n \times n$  è sufficiente limitarsi a studiare le curva di Pareto di tutte le sotto-bimatrici  $2 \times 2$  [15], inizialmente si fornirà una semplice ma esaustiva classificazione dell'insieme immagine delle funzioni di utilità di un generico gioco  $2 \times 2$  e da essa si ricaverà una classificazione di tutte le possibilità curve di Pareto. Successivamente verranno descritti gli algoritmi che, sfruttando la classificazione precedentemente ottenuta, confrontano le diverse curve di Pareto di ogni sotto-bimatrice  $2 \times 2$  di una generica bimatrice per calcolare

la curva di Pareto complessiva del gioco originale. Infine verrà analizzata la complessità computazionale degli algoritmi proposti, e discusso il risultato delle prove effettuate.

## Struttura

Il documento è strutturato come segue:

- **Capitolo 1: Preliminari e stato dell'arte**

Introduce i concetti, le ipotesi, le definizioni ed i modelli che costituiscono l'ossatura fondamentale della Tdg, corredati con diversi esempi. Vengono illustrati anche alcuni dei concetti di soluzione più rilevanti e le relazioni tra di essi. Infine, dopo aver introdotto gli strumenti per lo studio della complessità computazionale, viene valutato lo stato dell'arte dell'area Agt, in particolare per la computazione di soluzioni Pareto efficienti.

- **Capitolo 2: La Curva di Pareto in giochi  $2 \times 2$**

Viene fornita una classificazione completa dei giochi  $2 \times 2$  sulla base delle caratteristiche geometriche dell'insieme immagine delle funzioni di utilità dei due giocatori. Successivamente sono derivate tutte le possibili forme della relativa Curva di Pareto, fornendo anche in questo caso una classificazione esaustiva. Viene poi descritto l'algoritmo per il calcolo della Curva di Pareto in un gioco  $2 \times 2$ .

- **Capitolo 3: La Curva di Pareto in giochi  $n \times n$**

Sulla base del lavoro illustrato nel capitolo precedente, vengono descritti ed analizzati gli algoritmi per calcolare la Curva di Pareto di un gioco generico a 2 giocatori in forma normale, illustrando i risultati dei test effettuati.

- **Conclusioni**

Analizza il lavoro svolto, i risultati ottenuti ed introduce spunti per futuri approfondimenti.

# Capitolo 1

## Preliminari e stato dell'arte

### 1.1 Modelli di gioco

La Teoria dei giochi è stata recentemente definita "Interactive Decision Theory" [1]. Questa definizione rappresenta in maniera sintetica la collocazione della Teoria dei giochi, rispetto a discipline più classiche come Matematica, Economia e Ricerca operativa: la Teoria dei giochi mira a individuare decisioni ottimali, in presenza di più giocatori (agenti). Non stupisce quindi sapere che il contributo iniziale alla Tdg moderna, sia stato fornito in un lavoro di ricerca principalmente economico [17]. La prima domanda che potrebbe sorgere spontaneamente ad un neofita che si avvicina alla disciplina potrebbe essere: «che cosa si intende per gioco?». Ognuno può immaginare, con quanto detto finora, che sia in generale una situazione in cui più giocatori interagiscono in qualche modo. Pur non essendo un'asserzione errata, per porre le fondamenta di una teoria è necessario possedere astrazione e fornire una *rappresentazione* del concetto che vogliamo esprimere. La rappresentazione, nel nostro caso matematica, risulta determinante in quanto può enfatizzare o rendere invisibili alcune proprietà del concetto considerato.

Cominciamo quindi a distinguere 2 grossi filoni che rappresentano un gioco in maniera distinta:

- Teoria *cooperativa*: ogni giocatore può decidere di cooperare con altri (i.e. gli accordi sono vincolanti) e formare *coalizioni* in cui l'interesse di ogni membro non è esclusivamente personale, ma è anche comunitario per la coalizione.
- Teoria *non cooperativa*: l'interesse di ognuno dei giocatori è esclusivamente personale, cioè il soddisfacimento o il non soddisfacimento dell'interesse degli altri giocatori è totalmente irrilevante. Con questa premessa si deduce che gli accordi tra giocatori, non sono vincolanti.

Queste due teorie hanno avuto sviluppi storici indipendenti e tuttora rimangono di fatto teorie diverse, con modelli e assunzioni distinti. Questo non significa che nelle

applicazioni si debba utilizzare una o l'altra teoria in maniera mutuamente esclusiva: al contrario sono frequenti le situazioni che possono essere rappresentate sfruttando la teoria cooperativa o quella non cooperativa, ottenendo due modelli differenti della situazione che volevamo descrivere. Allo stesso modo sono frequenti casi in cui per ottenere una rappresentazione completa del fenomeno potrebbe essere necessario ricorrere ad entrambe le teorie (e.g. una partita di pallacanestro).

Nel seguito ci si concentrerà illustrando le basi della teoria *non cooperativa*, in quanto la classe principale di giochi considerata in questa tesi costituisce una rappresentazione tipicamente non cooperativa.

### 1.1.1 Assunzioni

Prima di procedere nella descrizione formale delle rappresentazioni che è possibile utilizzare per descrivere un gioco, è necessario chiarire quali sono le assunzioni fondamentali sui giocatori che prendono parte al gioco. Nel caso non cooperativo valgono le seguenti due ipotesi sui giocatori:

- *Razionalità*: ogni giocatore è razionale, cioè se è a conoscenza del meccanismo di gioco e dei valori di utilità degli altri giocatori può *ragionare* arrivando a concludere quale comportamento utilizzeranno gli altri giocatori, e scegliendo il proprio di conseguenza. Con questa ipotesi stiamo anche assumendo che ogni giocatore sappia ordinare gli esiti possibili del gioco secondo le sue preferenze, e che conosca le leggi della probabilità.
- *Egoismo*: ogni giocatore ha l'obiettivo di massimizzare il proprio tornaconto personale, non tenendo in considerazione quello altrui. Se rompere un accordo con un altro giocatore fosse conveniente, qualsiasi agente egoista lo farebbe. Pertanto non si può assumere che gli accordi siano vincolanti.

Stabilito questo, possiamo illustrare una prima forma in cui è possibile rappresentare un gioco.

### 1.1.2 Giochi in forma strategica

**Definizione** *Gioco in forma strategica*

Un gioco in forma normale (o strategica) è una tupla:

$$(N, A, V) \tag{1.1}$$

Dove:

$$N = \{1, 2, \dots, n\}$$

è l'insieme di  $n$  agenti (o giocatori),

$$A = A_1 \times A_2 \times \dots \times A_n$$

In cui  $A_i$  è l'insieme delle *azioni* disponibili all'agente  $i$ , chiamate anche *strategie pure*, e

$$V = (u_1, u_2, \dots, u_n)$$

Dove  $u_i : A \rightarrow \mathbb{R}$  è una *funzione di utilità* a valori reali, chiamati *payoff* per l'agente  $i$ .

Questo tipo di modello assume solitamente che le azioni dei giocatori vengano eseguite *contemporaneamente*. Per poter maneggiare agevolmente la tupla (1.1) si utilizza spesso una rappresentazione equivalente in forma matriciale. La descrizione in forma matriciale risulta particolarmente maneggevole nel caso di giochi a 2 giocatori. In questo caso parliamo di giochi *bimatrice*, in quanto ogni giocatore è dotato di una matrice. L'azione che il primo giocatore deve compiere è quella di selezionare una riga della bimatrice, il secondo deve specularmente selezionare una colonna. Il generico elemento della bimatrice è una coppia di numeri reali che prende il nome di *vettore di payoffs puri*. Esso rappresenta rispettivamente l'utilità ottenuta (o payoff) dal primo e dal secondo giocatore, a valle delle rispettive azioni. Una volta che i giocatori hanno effettuato le proprie azioni, l'esito del gioco (identificato con l'elemento della bimatrice selezionato) prende il nome di *esito*.

Formalmente:

**Definizione** *Gioco bimatrice*

Un gioco in forma strategica a 2 giocatori è un gioco bimatrice, cioè una tupla  $(A, B)$  con  $A, B$  matrici  $n \times m$  a valori reali. In cui le  $n$  righe sono le azioni del giocatore riga, e le  $m$  colonne quelle del giocatore colonna.

Per rendere concreta questa definizione illustriamo qualche gioco rappresentabile con una bimatrice.

**Esempio** *Sasso, carta, forbici*

Come ben noto, ogni giocatore ha 3 mosse a propria disposizione. Una volta che entrambi i giocatori hanno giocato la loro scelta, i possibili esiti dal punto di vista di uno dei giocatori sono 3: vittoria, sconfitta o pareggio. Su questi, è stabilito un ovvio ordinamento di preferenza che dovrà essere preservato nei valori di utilità che li rappresentano. Possiamo anche ragionevolmente assumere che le utilità dei giocatori siano le stesse, ad esempio 1 in caso di vittoria, -1 in caso di sconfitta e 0 in caso di pareggio. Infine assumiamo che la prima riga (colonna) corrisponda a giocare sasso, la seconda carta e la terza forbici. Sasso, carta, forbici in forma strategica diventa una bimatrice asimmetrica quadrata di ordine 3:

Tabella 1.1: Bimatrice di Sasso, carta, forbici.

	S	C	F
S	(0, 0)	(-1, 1)	(1, -1)
C	(1, -1)	(0, 0)	(-1, 1)
F	(-1, 1)	(1, -1)	(0, 0)

**Esempio** *Il dilemma del prigioniero*

Due persone sono sospettate di essere i responsabili di un grave crimine. Il giudice fa loro la seguente proposta: «per il crimine di cui vi ritengo colpevoli, posso condannarvi a 7 anni di carcere. Ad ogni modo se entrambi confessate, terrò in considerazione la cosa e vi condannerò soltanto a 5 anni. Se soltanto uno di voi confessa la responsabilità di entrambi (e l'altro no) colui che ha confessato sarà libero, mentre l'altro sconterà interamente la pena di 7 anni. Infine se nessuno di voi confessa, condannerò ognuno ad 1 anno di prigione perchè non ho abbastanza prove per dimostrare che siete colpevoli davanti ad una giuria (anche se sono sicuro che lo siate), ma posso provare che siete colpevoli di un crimine minore.». Assumiamo che la prima riga (colonna) corrisponda all'azione «confesso» e la seconda a «non confesso». La bimatrice del gioco risulta:

Tabella 1.2: Bimatrice del dilemma del prigioniero

	C	NC
C	(-5, -5)	(0, -7)
NC	(-7, 0)	(-1, -1)

**Esempio** *La battaglia dei sessi*

Una coppia deve decidere come passare la serata. Il marito vorrebbe andare a vedere una partita di football mentre la moglie vorrebbe andare al cinema. Naturalmente entrambi amano stare insieme, ma le loro prime scelte sono

differenti. Assumiamo quindi che il marito selezioni una riga, e la moglie una colonna della bimatrice. La prima riga (colonna) significa andare a vedere la partita di football, la seconda andare al cinema.

Tabella 1.3: Bimatrice della battaglia dei sessi.

	F	C
F	(3, 2)	(1, 1)
C	(-1, -1)	(2, 3)

L'esempio Sasso, carta, forbici costituisce un esempio di una classe di giochi bimatrice di indubbia importanza:

**Definizione** *Gioco a somma zero*

Un gioco bimatrice si dice a somma zero se  $A + B = 0$ .

Dovrebbe essere chiaro a questo punto cosa significhi rappresentare un gioco in forma strategica. Sarà poi saltato all'occhio come la scelta esatta delle funzioni di utilità dei giocatori sia il più delle volte arbitraria: quello che è importante è l'ordinamento relativo tra i valori di utilità associati ai possibili esiti del gioco. Spostiamo ora l'attenzione sui giocatori: è stato più volte rimarcato il fatto che i giocatori sono razionali, cioè che possono ragionare e, assumendo che il meccanismo di gioco e le funzioni di utilità siano note, dedurre il comportamento che avranno gli avversari. Per specificare nella prossima sezione cosa s'intende con queste parole, è necessario anzitutto definire il concetto di *strategia*:

**Definizione** *Strategia*

Una strategia, indicata con il vettore  $\mathbf{x}_i$  specifica con quale probabilità il giocatore  $i$  giocherà le azioni a lui disponibili. Cioè  $|\mathbf{x}_i| = |A_i|$  e, sia  $a \in A_i$ , valgono:

$$\begin{aligned} \forall i, a \quad x_{i,a} &\geq 0 \\ \forall i \quad \sum_{a \in A_i} x_{i,a} &= 1 \end{aligned}$$

Una strategia si dice pura se  $\mathbf{x}_i$  ha un elemento uguale ad 1 e tutti gli altri nulli. Quando questo non avviene si parla di strategie *miste*. Nel caso in cui ogni elemento di  $\mathbf{x}_i$  sia strettamente positivo si parlerà di strategie *completamente miste* (*fully mixed*). Un vettore  $n$ -dimensionale di strategie (una per ogni giocatore) prende il nome di *profilo di strategie*. Ricordiamo anche le seguenti definizioni:

**Definizione** *Simplesso*

Sia  $p \in \mathbb{N}$ . Indichiamo il  $(p - 1)$ -simplesso

$$\Delta_p = \left\{ \mathbf{z} \in \mathbb{R}^p : \forall i \in [p], z_i \geq 0, (\mathbf{1})^T \mathbf{z} = 1 \right\}$$

Una strategia mista è quindi definita su un semplice.

**Definizione** *Supporto di una strategia*

Per qualsiasi strategia  $\mathbf{x}_i$  il supporto  $S_{x_i}$  è l'insieme delle coordinate di  $\mathbf{x}_i$  con valore strettamente positivo:

$$S_x = \{j : x_j \in \mathbf{x}_i, x_j > 0\}$$

**Definizione** *Utilità attesa (gioco bimatrice)*

Sia  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$  il profilo di strategie dei 2 giocatori:  $\mathbf{u}(\mathbf{x}) = (u_{1e}, u_{2e})$  con  $u_{1e} = \mathbf{x}_1^T A \mathbf{x}_2$  e  $u_{2e} = \mathbf{x}_1^T B \mathbf{x}_2$  è la coppia di utilità attese rispettivamente del primo e del secondo giocatore.

### 1.1.3 Giochi in forma estesa

La rappresentazione in forma strategica, presenta determinate caratteristiche che la rendono più o meno adatta a rappresentare alcuni giochi. Sebbene sia la più frequentemente utilizzata, essa non è l'unica possibile. Una modellazione alternativa è quella dei *giochi in forma estesa*, in cui i giocatori giocano non più contemporaneamente ma *sequenzialmente*, anche più volte nello stesso gioco. Comunemente vengono denotati attraverso l'ausilio di un albero decisionale: ogni nodo (detto *nodo decisionale*) rappresenta una scelta che un determinato giocatore è chiamato a compiere in quel momento del gioco. Esso deve scegliere quale azione giocare tra quelle a lui disponibili in quel nodo, cioè quale ramo percorrere. Conseguentemente l'origine dell'albero costituisce l'inizio del gioco e le foglie la fine dello stesso. Sulle foglie, analogamente a quanto accade nei giochi in forma strategica, vengono indicati i payoffs di tutti i giocatori. I giochi in forma estesa si possono dividere poi in due classi:

- Giochi ad informazione perfetta: ogni giocatore osserva tutte le azioni giocate dagli altri giocatori, oltre che del meccanismo di giochi e di tutte le funzioni di utilità. Rientrano in questa categoria ad esempio la Dama cinese, o il Go.
- Giochi ad informazione imperfetta: non tutte le azioni giocate dagli avversari sono osservabili. Un esempio di questi tipo di giochi è ad esempio il Poker.

## 1.2 Concetti di soluzione

Una volta rappresentato il gioco d'interesse nella rappresentazione più opportuna, dobbiamo esprimere matematicamente cosa intendiamo per *razionalità* dei giocatori, ed in che modo essi possano decidere quali strategie giocare durante il gioco. Iniziamo quindi con illustrare il primo criterio di dominanza:

**Definizione** *Dominanza Strategica*: un qualsiasi giocatore  $i$  non sceglierà una strategia  $\bar{x}_i$  se è disponibile un'altra strategia  $\hat{x}_i$  che gli consenta di guadagnare *strettamente* di più, indipendentemente dalla scelta degli altri giocatori.

Questo criterio è il primo strumento con cui è possibile risolvere i giochi in forma strategica, e porta ad identificare delle soluzioni del gioco che sono, rispetto alle nostre assunzioni, incontrovertibilmente razionali.

**Esempio** *Il dilemma del prigioniero*

Consideriamo la bimatrice illustrata nella Tabella 1.2. Poichè  $-5 > -7$  e  $0 > -1$ , qualsiasi sia la scelta del secondo giocatore, al primo conviene sempre confessare. In altre parole entrambi i giocatori si rendono conto che in virtù del criterio di dominanza strategica, il primo giocatore non giocherà mai la seconda riga e conseguentemente la bimatrice del gioco può essere trasformata nella seguente:

Tabella 1.4: Bimatrice del dilemma del prigioniero: eliminazione strategia dominata.

	C	NC
C	$(-5, -5)$	$(0, -7)$

Per lo stesso principio a questo punto il secondo giocatore non ha interesse nel non confessare. La seconda colonna risulta anch'essa una strategia dominata. Eliminandola otteniamo:

Tabella 1.5: Bimatrice del dilemma del prigioniero: esito.

	C
C	$(-5, -5)$

Cioè l'esito del gioco è inequivocabile: entrambi i giocatori confesseranno, perchè indipendentemente dalle scelte dell'altro è sempre più conveniente confessare. Questo dimostra come, in situazioni in cui gli agenti in gioco hanno interessi contrapposti, la razionalità individuale determina un esito che può essere tutt'altro che ottimale. Infatti i due giocatori scontreranno una pena di 5 anni, quando tra gli esiti possibili c'era la pena di un solo anno ciascuno. Questo esito, preferibile per entrambi rispetto alla pena di 5 anni, non è individualmente razionale in quanto se un giocatore non confessasse, l'altro comunque avrebbe interesse nel confessare, evitando il carcere anzichè scontare 1 anno di detenzione.

In Sasso, carta, forbici e La battaglia dei sessi non esistono azioni dominate. Considerate le situazioni di gioco questo risulta abbastanza ragionevole ed è un segnale

del fatto che il criterio di dominanza strategica non è sufficiente a determinare una possibile soluzione di un gioco.

### 1.2.1 Equilibrio di Nash

**Definizione** *Equilibrio di Nash (NE)*

Considerando un gioco non cooperativo a 2 giocatori in forma normale<sup>1</sup> (i.e. un gioco bimatrice) un Equilibrio di Nash è un profilo di strategie  $(\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2)$  tale che:

$$\begin{aligned}\forall \mathbf{x}_1 \bar{\mathbf{x}}_1^T A \bar{\mathbf{x}}_2 &\geq \mathbf{x}_1^T A \bar{\mathbf{x}}_2 \\ \forall \mathbf{x}_2 \bar{\mathbf{x}}_1^T B \bar{\mathbf{x}}_2 &\geq \bar{\mathbf{x}}_1^T B \mathbf{x}_2\end{aligned}$$

Questo concetto, centrale nella Tdg, risulta particolarmente interessante. Supponiamo, in una situazione in cui il criterio di dominanza precedentemente esposto non fosse d'aiuto, che qualcuno suggerisca ai giocatori di giocare l'Equilibrio di Nash  $(\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2)$ . Dobbiamo aspettarci un possibile rifiuto da parte dei giocatori di utilizzare le strategie suggerite? La risposta è negativa: ciascuno dei giocatori, *assumendo che tutti gli altri giochino la strategia suggerita*, non ha interesse nel deviare dalla strategia a lui consigliata. D'altra parte è ragionevole assumere che nessun'altro devii, per lo stesso motivo. Notiamo poi che questo concetto di soluzione è una generalizzazione delle soluzioni ottenute con il criterio di dominanza. Infatti ogni soluzione di questo tipo è un Equilibrio di Nash.

Ricordiamo anche:

**Teorema** Ogni gioco finito ammette un Equilibrio di Nash in strategie miste.

**Esempio** *La battaglia dei sessi*

Facendo riferimento alla bimatrice illustrata nella Tabella 1.3 è possibile notare come esistano 2 Equilibri di Nash in strategie pure: uno associato al payoff  $(3, 2)$  e l'altro associato al payoff simmetrico  $(2, 3)$ . Questo è intuitivo: la battaglia dei sessi rappresenta un gioco in cui esistono due equilibri che risultano soddisfacenti, cioè quelle situazioni in cui marito e moglie rimangono insieme. In questo senso è un *coordination game*, cioè i giocatori devono coordinarsi affinché l'esito del gioco con strategie pure, sia uno degli equilibri di Nash esposti. Esiste un altro equilibrio di Nash in strategie miste associato al profilo di strategie in cui ogni giocatore gioca la sua prima scelta (football per il marito e cinema per la moglie) con probabilità  $\frac{4}{5}$ .

<sup>1</sup>Viene fornita la definizione per due giocatori per semplicità, ma è estendibile al caso con  $n$  giocatori.

**Esempio** *Sasso, carta, forbici*

In questo caso, analizzando la bimatrice illustrata nella Tabella 1.1 non esistono equilibri di Nash in strategie pure. Per il teorema sopra esposto sappiamo però che ne esiste almeno uno in strategie miste. Si può dimostrare con qualche semplice calcolo che in questo caso l'unico equilibrio di Nash risulta quello in cui entrambi i giocatori giocano con ugual probabilità  $0, \bar{3}$  le mosse a loro disposizione.

**1.2.2 Pareto Efficienza**

Come anticipato nella sezione precedente, nelle applicazioni non sempre è chiaro se utilizzare un modello cooperativo o non cooperativo. Finora sono stati presentati solutions concepts che rientrano chiaramente nel caso non cooperativo: essi infatti sono espressione delle ipotesi di questa teoria: razionalità ed egoismo individuale. Vogliamo ora definire una proprietà che conferisce alle soluzioni che la possiedono un interesse di carattere sociale, oltre che individuale. Tale proprietà prende il nome di Pareto efficienza ed è stata formulata agli inizi del XX secolo (ben prima della nascita della Tdg moderna) da Vilfredo Pareto, in un trattato di economia [28].

**Definizione** *Pareto efficienza*

Un profilo di strategie  $\mathbf{x}$  (analogamente anche  $\mathbf{u}(\mathbf{x})$ ) è Pareto efficiente se non esiste nessun altro profilo di strategie  $\mathbf{x}'$  (analogamente  $\mathbf{u}(\mathbf{x}')$ ) tale che  $\mathbf{u}(\mathbf{x}') \geq \mathbf{u}(\mathbf{x})$  e  $u_i(\mathbf{x}') > u_i(\mathbf{x})$  per qualche giocatore  $i$ , dove  $u_i$  è l'utilità attesa del giocatore  $i$ -esimo.

**Definizione** *Pareto efficienza debole*

Un profilo di strategie  $\mathbf{x}$  (analogamente anche  $\mathbf{u}(\mathbf{x})$ ) è debolmente Pareto efficiente se non esiste nessun altro profilo di strategie  $\mathbf{x}'$  (analogamente  $\mathbf{u}(\mathbf{x}')$ ) tale che  $\mathbf{u}(\mathbf{x}') > \mathbf{u}(\mathbf{x})$ .

**Definizione** *Curva di Pareto*

La Curva di Pareto è la collezione di  $\mathbf{u}(\mathbf{x})$  tali che  $\mathbf{x}$  è Pareto efficiente.

**Definizione** *Curva di Pareto debole*

La Curva di Pareto debole è la collezione di  $\mathbf{u}(\mathbf{x})$  tali che  $\mathbf{x}$  è debolmente Pareto efficiente.

**Definizione** *Pareto dominanza forte*

Un profilo di strategie  $\mathbf{x}$  (analogamente anche  $\mathbf{u}(\mathbf{x})$ ) è (fortemente) Pareto dominato dal profilo di strategie  $\mathbf{x}'$  (analogamente  $\mathbf{u}(\mathbf{x}')$ ) se  $\mathbf{u}(\mathbf{x}') > \mathbf{u}(\mathbf{x})$ .

**Definizione** *Pareto dominanza debole*

Un profilo di strategie  $\mathbf{x}$  (analogamente anche  $\mathbf{u}(\mathbf{x})$ ) è Pareto dominato debolmente dal profilo di strategie  $\mathbf{x}'$  (analogamente  $\mathbf{u}(\mathbf{x}')$ ) se  $\mathbf{u}(\mathbf{x}') \geq \mathbf{u}(\mathbf{x})$  e

$u_i(\mathbf{x}') > u_i(\mathbf{x})$  per qualche giocatore  $i$ , dove  $u_i$  è l'utilità attesa del giocatore  $i$ -esimo.

La Curva di Pareto è quindi l'insieme dei profili di strategie che non sono debolmente Pareto dominati, mentre l'insieme dei profili di strategie che non sono fortemente Pareto dominati costituiscono la Curva di Pareto debole.

In altre parole una soluzione Pareto efficiente è una soluzione per cui non è possibile far guadagnare di più un certo giocatore (variando il profilo di strategie, e quindi la soluzione) senza che non ce ne sia un altro che guadagni di meno. Per questo motivo la Pareto efficienza è una proprietà estremamente importante nei cosiddetti problemi di *Bargaining* (contrattazione). Le soluzioni per questo tipo di problemi infatti, devono essere necessariamente Pareto efficienti per poter essere razionali.

**Esempio** *Il dilemma del prigioniero*

La Curva di Pareto è composta dai profili di strategie in cui (almeno) uno dei giocatori non confessa, e l'altro gioca una strategia mista qualsiasi. Profili di questo tipo infatti non possono essere migliorati in modo da portare guadagni maggiori ad uno dei giocatori senza far peggiorare l'altro. Il payoff puro  $(-1, -1)$  costituisce un interessante soluzione Pareto efficiente, tanto che diversi solution concepts di problemi di contrattazione la selezionerebbero.

**Esempio** *Sasso, carta, forbici*

In questo caso qualsiasi profilo di strategia risulta Pareto efficiente.

**Esempio** *La battaglia dei sessi*

La Curva di Pareto è costituita unicamente da 2 punti, corrispondenti agli esiti del gioco in cui la coppia passa la serata insieme. Non esistono profili di strategie miste nella Curva di Pareto in quanto qualsiasi possibile randomizzazione creerebbe la possibilità per la coppia di non stare insieme, che è un peggioramento per entrambi.

### 1.2.3 Relazioni

Potrebbe sorgere quindi spontanea la domanda: *che relazione c'è tra la Curva di Pareto e gli equilibri di Nash?*

La relazione tra questi due insiemi non è univocamente definita, ma dipende dal gioco che si sta considerando. Per questo motivo riportiamo per ognuno degli esempi presentato in precedenza, la relazione che intercorre tra l'insieme di soluzioni Pareto efficienti (che indicheremo con  $P$ ) e l'insieme degli equilibri di Nash (che indicheremo con  $N$ ).

**Esempio** *Il dilemma del prigioniero*

In questo gioco, l'egoismo individuale determina un'unica soluzione razionale

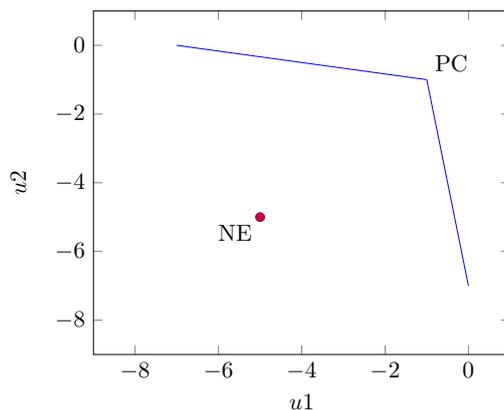


Figura 1.1: Dilemma del prigioniero: Curva di Pareto e NE.

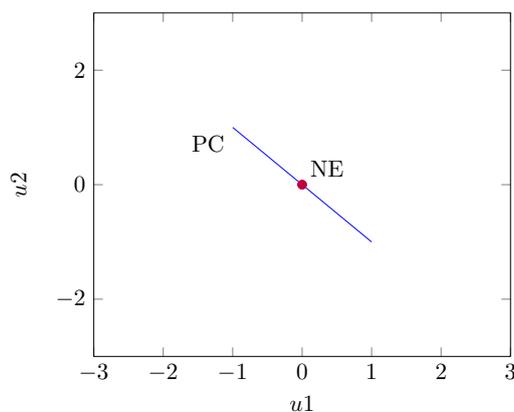


Figura 1.2: Sasso, carta, forbici: Curva di Pareto e NE.

che costituisce l'unico equilibrio di Nash e che è palesemente Pareto dominato. in Figura 1.1 rappresentiamo l'unico equilibrio di Nash e la Curva di Pareto. Abbiamo quindi che:

$$N \cap P = \emptyset$$

**Esempio** *Sasso, carta, forbici*

Come accennato in precedenza, ogni profilo di strategie risulta essere Pareto efficiente. Anche in questo caso esiste un unico equilibrio di Nash (Figura 1.2) cioè:

$$N \subset P$$

**Esempio** *La battaglia dei sessi*

La situazione è rappresentata nella Figura 1.3. Le soluzioni Pareto efficienti in questo caso sono unicamente associate a profili di strategie pure, e sono entrambi equilibri di Nash. Esiste però anche un terzo equilibrio di Nash

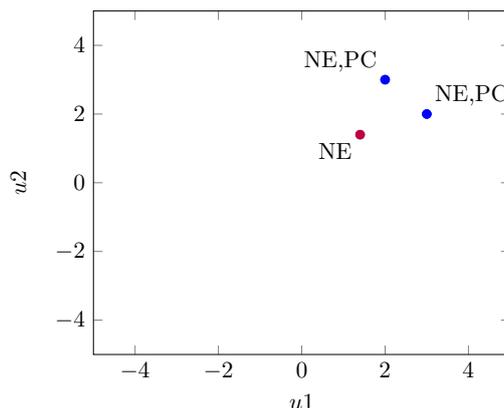


Figura 1.3: La battaglia dei sessi : Curva di Pareto e NE.

(con un profilo di strategie completamente miste) che non risulta essere Pareto efficiente. Specularmente al caso precedente abbiamo quindi:

$$P \subset N$$

Abbiamo visto finora che sono possibili tre tipi di relazioni tra la Curva di Pareto e l'insieme degli equilibri di Nash. Il caso in cui questi coincidano completamente è banalmente verificabile quando, ad esempio, in un gioco a 2 giocatori entrambi i giocatori sono totalmente indifferenti al gioco (i.e. la bimatrice è identicamente nulla). L'ultimo caso che non è stato per ora mostrato è quello in cui i due insiemi hanno un'intersezione non vuota, ma nessuno è contenuto nell'altro. Il prossimo esempio dimostrerà che è un'eventualità possibile.

**Esempio** *Una serata di gala*

Una coppia si sta preparando per partecipare ad una serata di gala. Entrambi nell'armadio hanno 2 vestiti a disposizione tra cui scegliere. Il marito può scegliere tra uno smoking ed un tight, mentre la moglie ha a disposizione un bellissimo abito da sera che si abbina perfettamente allo smoking del marito, ed un tailleur elegante, ma meno appariscente. Poichè a questa serata ci saranno tante persone importanti la moglie, che notoriamente è molto puntigliosa, vuole essere sicura di non sfigurare di fronte agli altri invitati. Pertanto ha un unico desiderio: presentarsi con l'abito da sera, e che il marito sia in smoking. Qualsiasi altra combinazione dei vestiti scelti, le sarebbe totalmente indifferente, perchè la renderebbe ugualmente insoddisfatta. Il marito dal canto suo ha ben altre opinioni: in qualsiasi contesto e situazione preferisce il tight rispetto allo smoking. Vorrebbe inoltre che la moglie mettesse il tailleur piuttosto che l'abito da sera: con l'abito sarebbe troppo bella e attirerebbe molti sguardi, facendolo diventare geloso. Con una eccezione: essendo a conoscenza del desiderio della moglie, se proprio fosse obbligato a mettere lo smoking, a

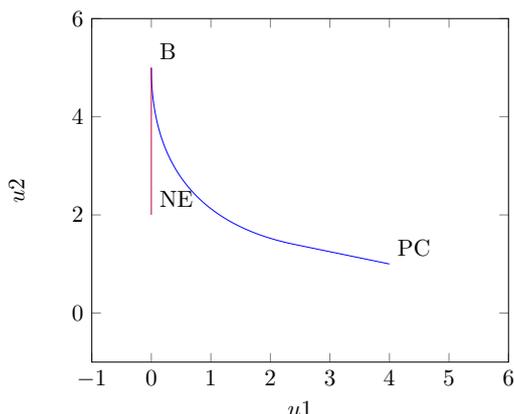


Figura 1.4: La serata di gala : Curva di Pareto e NE.

quel punto sarebbe meno insoddisfatto se la moglie mettesse l'abito da sera. Attirerebbe ancora molti sguardi, ma vedere la moglie contenta allieverebbe la scomodità dello smoking. Il gioco bimatrice, in cui il marito sceglie una colonna e la moglie una riga, è riportato nella Tabella 1.6:

Tabella 1.6: Bimatrice della serata di gala.

	S	T
T	(0,0)	(0,5)
A	(4,1)	(0,2)

Utilizzando il criterio delle strategie dominanti sappiamo per certo che il marito indosserà sicuramente il tight, ma a quel punto alla moglie qualsiasi scelta sarà totalmente indifferente. Pertanto esistono un'infinità di equilibri di Nash in cui il marito indossa il tight. La Curva di Pareto invece è meno evidente da individuare, anche se i payoff (4, 1) e (0, 5) ragionevolmente sembrano Pareto efficienti. La situazione è illustrata nella Figura 1.4. La relazione tra  $N$  e  $P$  è quindi:

$$N \cap P \neq \emptyset, N \not\subseteq P, P \not\subseteq N$$

Con quest'ultimo esempio abbiamo dimostrato che la relazione intercorrente tra la Curva di Pareto e l'insieme degli equilibri di Nash, non ha vincoli e dipende esclusivamente dal gioco considerato.

## 1.3 Complessità computazionale dei solution concepts

### 1.3.1 Cenni di complessità computazionale

Per completare la rassegna degli elementi di base che sono indispensabili per comprendere il lavoro di questo documento, è necessario fare qualche cenno di una

branca dell'informatica teorica. Con il termine *complessità computazionale* s'intende la quantità di risorse, in termini di tempo o memoria, necessarie ad un algoritmo per poter svolgere il proprio compito, in relazione alle dimensioni del problema considerato. Naturalmente questo tipo di analisi viene svolta su un algoritmo e non su un programma già codificato e destinato ad una specifica architettura. Questo significa che la quantità di memoria utilizzata, intesa come memoria necessaria per memorizzare i risultati parziali o definitivi dell'algoritmo (quindi nel caso delle architetture moderne stiamo parlando di memoria centrale, od al più quella del disco rigido) viene misurata come numero di celle di memoria utilizzate (per qualche grandezza della cella) mentre il tempo speso dall'algoritmo viene misurato come numero di operazioni elementari eseguite. Questo è utile da un punto di vista teorico per avere un'idea, prima di codificarlo, delle performance di un algoritmo al variare della grandezza dell'istanza del problema preso in input. Il tipo di analisi che si compie è quindi di tipo *asintotico*: ci si dimentica delle costanti moltiplicative o additive e si classifica l'algoritmo considerato rispetto al termine che, al crescere delle dimensioni dell'istanza, assume più peso nella funzione. In questo modo è possibile dimenticarsi i termini «più leggeri» e classificare l'algoritmo in una classe generica.

Una volta definito il concetto di complessità per uno specifico algoritmo, è naturale domandarsi qual'è il miglior algoritmo (in termini di complessità, solitamente temporale) per risolvere un certo problema. Ci si riferirà quindi alla *complessità di un problema* sottointendendo la complessità del miglior algoritmo per risolvere il problema considerato. In questo senso l'informatica teorica differenzia due classi di problemi distinti, a seconda della loro natura:

- Problemi decisionali: sono quei problemi che ammettono una output binario (sì o no).
- Problemi di ricerca: l'algoritmo deve ricercare una soluzione al problema e non deve limitarsi a fornire una risposta binaria, ma in caso di esito positivo (soluzione trovata) la deve anche esibire.

A seconda della tipologia del problema considerato, una volta analizzata la sua complessità temporale, lo si potrà classificare nella classe più appropriata. Per quanto riguarda i problemi decisionali, le due classi più importanti sono:

- $\mathcal{P}$  : costituisce la classe dei problemi decisionali che possono essere risolti in un tempo polinomiale rispetto alle dimensioni dell'istanza del problema. In altre parole sia  $n$  la dimensione dell'istanza, questo tipo di problemi può essere risolto ottenendo una risposta «sì/no» in  $\mathcal{O}(n^c)$  con  $c$  costante.
- $\mathcal{NP}$  : costituisce la classe dei problemi decisionali in cui, specificato il problema e una presunta soluzione di tipo «sì», è possibile produrre un *certificato* di veridicità di tale soluzione in un tempo  $\mathcal{O}(n^c)$  con  $c$  costante.

Mentre la classe  $\mathcal{P}$  specifica la complessità per la risoluzione del problema, deve essere chiaro che nessuna informazione viene fornita sull'algoritmo di risoluzione di un problema in  $\mathcal{NP}$  : le uniche informazioni a nostra disposizione riguardano la complessità di un algoritmo che verifichi la veridicità di una soluzione positiva. Definiamo anche la classe  $\text{co-}\mathcal{NP}$ : un problema decisionale  $P$  appartiene a  $\text{co-}\mathcal{NP}$  se e solo se il suo complementare  $P^c$  (per la definizione di problema complementare si veda [26]) si trova in  $\mathcal{NP}$ . Analogamente per i problemi di ricerca esistono le classi.

- $\mathcal{FP}$  : costituisce la classe dei problemi di ricerca che possono essere risolti in tempo polinomiale rispetto alle dimensioni dell'istanza del problema.
- $\mathcal{FNP}$ : è la classe dei problemi di ricerca, in cui specificato il problema ed una presunta soluzione è possibile produrre una certificazione di correttezza della soluzione un tempo polinomiale.

È sempre possibile poi associare ad ogni problema di ricerca, l'analoga versione decisionale. Mentre è evidente che il problema di ricerca è almeno tanto complesso quanto quello decisionale, non vi sono garanzie che avvenga il contrario. Per completare la rassegna dei concetti chiave forniti dall'informatica teorica è necessario definire il concetto di *riduzione*:

**Definizione** *Riduzione*

Un problema decisionale  $P1$  è riducibile ad un altro problema  $P2$  se per ogni istanza  $p_1$  di  $P1$ , si può costruire in tempo polinomiale una corrispondente istanza  $p_2$  di  $P2$  tale che, data una soluzione di  $p_2$  sia possibile ricavare sempre in tempo polinomiale, una soluzione di  $p_1$ . Se  $P1$  è riducibile a  $P2$  scriviamo:  $P1 \propto P2$ .

La definizione fornita implica che se  $P1 \propto P2$ , allora la complessità temporale di  $P2$  è maggiore o uguale a quella di  $P1$ . In altre parole  $P2$  non può essere più semplice (in termine di complessità temporale) di  $P1$ . Questa definizione è necessaria per introdurre le seguenti due classi di problemi, centrali nell'informatica teorica:

**Definizione**  $\mathcal{NP}$ -completezza

Un problema decisionale  $P$  si dice  $\mathcal{NP}$ -completo se e solo se appartiene ad  $\mathcal{NP}$  e ogni altro problema in  $\mathcal{NP}$  è riducibile ad esso in tempo polinomiale, cioè  $\forall P' \in \mathcal{NP} P' \propto P$ .

In altre parole i problemi  $\mathcal{NP}$ -completi, rappresentano i problemi della classe  $\mathcal{NP}$  più difficili da risolvere, cioè che presentano la massima complessità temporale tra tutti i problemi della classe  $\mathcal{NP}$ . La classe di problemi di questo tipo viene indicata con  $\mathcal{NPC}$ . Spingendosi al di fuori dei confini della classe  $\mathcal{NP}$  possiamo considerare problemi ancora più difficili:

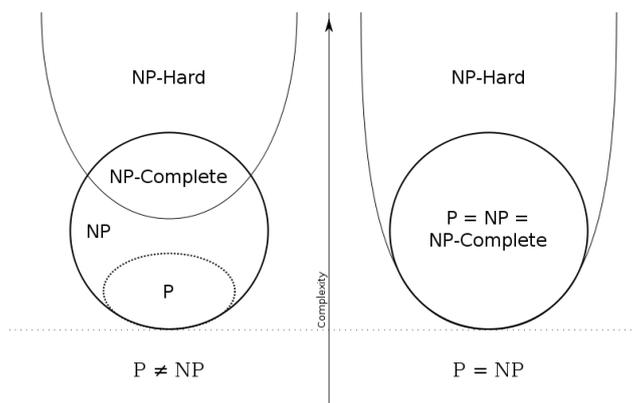


Figura 1.5: Diverse relazioni tra le classi di complessità.

**Definizione  $\mathcal{NP}$ -hardness**

Un problema decisionale  $P$  si dice  $\mathcal{NP}$ -difficile (o  $\mathcal{NP}$ -hard) se ogni problema  $\mathcal{NP}$ -completo è riducibile ad esso in tempo polinomiale, cioè  $\forall P' \in \mathcal{NPC} P' \propto P$ .

Questa definizione non pone nessuna altra condizione sulla classe di  $P$ , pertanto i problemi  $\mathcal{NP}$ -difficili sono complessi almeno quanto quelli  $\mathcal{NP}$ -completi. L'insieme di questi problemi si indica con  $\mathcal{NPH}$ . Le relazioni tra queste classi è tuttora oggetto di ricerca nel campo dell'informatica teorica. La relazione intercorrente tra  $\mathcal{P}$  e  $\mathcal{NP}$  in particolare è talmente d'interesse che la domanda  $\mathcal{P} = \mathcal{NP}$ ? è uno dei *Millennium Problems* sanciti dal Clay Mathematics Institute. La Figura 1.5 riporta le principali classi di complessità dei problemi decisionali e le relazioni intercorrenti tra di esse, per entrambe le risposte alla domanda  $\mathcal{P} = \mathcal{NP}$ ?

Analogamente a quanto fatto con i problemi decisionali, si definiscono per i problemi di ricerca la classe dei problemi  $\mathcal{FNP}$ -completi ( $\mathcal{FNPC}$ ) e quelli  $\mathcal{FNP}$ -difficili ( $\mathcal{FNPH}$ )

**1.3.2 Complessità della ricerca di un equilibrio di Nash**

Le condizioni che caratterizzano un Equilibrio di Nash in un gioco non cooperativo in forma strategica possono essere espresse in una forma alternativa rispetto a quella precedentemente descritta. Focalizzandoci su giochi bimatrice, il problema di ricerca degli Equilibri di Nash  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$  può essere espresso come un problema costituito da vincoli non lineari:

$$\mathbf{x}_1, \mathbf{x}_2 \geq 0 \tag{1.2}$$

$$\mathbf{1}v_1 - A \cdot \mathbf{x}_2 \geq 0, \mathbf{1}v_2 - B \cdot \mathbf{x}_1 \geq 0 \tag{1.3}$$

$$\mathbf{x}_1^T \cdot (\mathbf{1}v_1 - A \cdot \mathbf{x}_2) = 0, \mathbf{x}_2^T \cdot (\mathbf{1}v_2 - B \cdot \mathbf{x}_1) = 0 \tag{1.4}$$

$$\mathbf{1}^T \cdot \mathbf{x}_1 = 1, \mathbf{1}^T \cdot \mathbf{x}_2 = 1 \tag{1.5}$$

Dove con  $v_i$  si è indicata l'utilità attesa dell'agente  $i$ . I vincoli (1.2) e (1.5) rappresentano il fatto che la strategia di ogni giocatore appartiene al relativo semplice. (1.3) ci dice che nessuna strategia pura dell'agente  $i$  può consentirgli di guadagnare più di  $v_i$  mentre il vincolo (1.4) afferma che ogni agente gioca unicamente azioni ottimali (*principio d'indifferenza*). Una volta fornita questa rappresentazione, possiamo considerare due problemi relativi agli Equilibri di Nash:

- La verifica che un profilo di strategie sia un Equilibrio di Nash (problema decisionale).
- La ricerca di un Equilibrio di Nash (problema di ricerca).

I maggiori contributi a queste problematiche sono forniti in [9, 11]. Poiché in questo documento vengono discussi giochi bimatrice, citeremo risultati e risultati di complessità riguardanti NE per giochi a 2 giocatori. Per quanto riguarda la verifica, è sufficiente sostituire nei vincoli il profilo di strategie fornito in input, e verificare se questi sono soddisfatti. Il problema è quindi, considerando come dimensione dell'istanza il numero di mosse dei giocatori, banalmente in  $\mathcal{P}$ . Il problema di ricerca invece, appare più complesso. Poiché possiamo verificare che un profilo di strategie sia effettivamente un Equilibrio di Nash richiede un tempo polinomiale, sicuramente il problema di ricerca apparterrà a  $\mathcal{FNP}$ . In effetti, anche senza raffinate analisi, possiamo dire di più: il teorema di Nash ci garantisce che (indipendentemente dal numero di giocatori) esista sempre almeno un equilibrio di Nash in strategie miste. È utile quindi definire la classe  $\mathcal{TFNP} \subset \mathcal{FNP}$  (Total function  $\mathcal{FNP}$ ) come la classe dei problemi appartenente a  $\mathcal{FNP}$ , che ammette sempre una soluzione. Questa classe non viene studiata direttamente ma si considerano suoi sottoinsiemi [11]. Le prossime definizioni sono necessarie per collocare il problema di ricerca di un Equilibrio di Nash in un'adeguata classe di complessità:

**Definizione** *Problema «fine della linea»*

Date due funzioni  $S$  e  $P$ , ognuna con un input ed un output di  $n$  bits, tale che  $P(0^n) = 0^n \neq S(0^n)$ , trova un input  $x \in \{0, 1\}^n$  tale che  $P(S(x)) \neq x$  o  $S(P(x)) \neq x \neq 0^n$ .

**Definizione** *Classe  $\mathcal{PPAD}$  (Polynomial Parity Argument for Directed Graph)*

La classe  $\mathcal{PPAD}$  è la classe di problemi di ricerca totali, riducibili in tempo polinomiale al problema «fine della linea».

In altre parole un problema totale appartenente alla classe  $\mathcal{PPAD}$  può essere rappresentato come un grafo (discreto) orientato, i cui vertici hanno al più grado entrante ed uscente 1, in cui è presente una sorgente (nota) ed esiste

- una funzione che fornito un nodo in input, restituisce in output il nodo successivo (ed il precedente).
- un nodo con grado uscente nullo (pozzo) che rappresenta la soluzione del problema.

in [11] e [9] si dimostra il seguente teorema.

**Teorema** Il problema di ricerca di un Equilibrio di Nash in un gioco bimatrice è  $PPAD$ -completo. Il numero di vertici del grafo del problema «fine della linea» a cui si può ridurre, cresce esponenzialmente con il numero di azioni dei giocatori.

Infine, è risaputo che  $PPAD \subseteq NP$  ( $PPAD \not\subseteq NPC$  a meno che  $NP = co-NP$ ) e si ritiene generalmente che  $PPAD \neq P$ . Nel caso peggiore quindi la complessità di individuare un Equilibrio di Nash è esponenziale nel numero delle azioni dei giocatori.

### 1.3.3 Complessità soluzioni Pareto efficienti

Come anticipato, la proprietà di Pareto efficienza ha radici ben più profonde di quelle della Tdg. In effetti questa proprietà è studiata in una categoria più generale di problemi, quelli di *ottimizzazione multi obiettivo* nella forma:

$$\begin{aligned} \max \mathbf{f}(\mathbf{x}) \\ s.t \mathbf{g}(\mathbf{x}) = \mathbf{b} \\ \mathbf{x} \geq \mathbf{0} \end{aligned}$$

In cui tutte le grandezze coinvolte, sono grandezze vettoriali. Una possibile soluzione a problemi in questa forma, è rappresentata da tutti e soli i punti Pareto efficienti di  $\mathbf{f}(\mathbf{x})$ , cioè da tutti quei punti che garantiscono il miglior trade-off tra le varie componenti della funzione vettoriale. I risultati noti in letteratura circa la complessità temporale degli algoritmi di ricerca delle soluzioni Pareto efficienti, hanno come punto di partenza questo tipo di problemi. In [16] gli autori mostrano che, nel caso generale (funzione obiettivo arbitraria) la ricerca di soluzioni esatte è un problema  $NP$ -difficile, ma persino la ricerca di soluzioni approssimate risulta essere in generale  $NP$ -difficile. In particolare definiscono dei concetti di soluzione approssimate e non, ed indagano quando questi possono essere individuati in tempo polinomiale. A differenza dei problemi in cui si sta minimizzando, il caso della massimizzazione risulta essere un problema difficile, a meno che  $P = NP$ . [20] propone una metodologia generale per l'approssimazione della Curva di Pareto per problemi di ottimizzazione multi criterio. Il metodo proposto si basa sull'invio di richieste ad un *constraint solver*, e garantisce un bound sulla distanza della soluzione approssimata rispetto alla Curva di Pareto. [27] mostra come, sotto alcune condizioni molto

generali, è possibile individuare in tempo polinomiale una curva che  $\epsilon$ -approssima la Curva di Pareto, per ogni  $\epsilon > 0$ . Vengono fornite anche le condizioni necessarie e sufficienti sotto le quali è possibile costruire tale curva in tempo polinomiale rispetto alle dimensioni dell'istanza e rispetto a  $\frac{1}{\epsilon}$ . [14] Propone un metodo per approssimare l'insieme dei punti non dominati, quando sia la funzione vettoriale che il feasible set sono convessi. Anche in questo caso il metodo fornisce un insieme di punti  $\epsilon$ -non dominati.

Un approccio alternativo alla questione viene fornito in [8]: classicamente gli algoritmi evolutivi (i.e. euristiche) che fanno uso di diverse metriche per trovare soluzioni nello spazio obiettivo (per problemi multi criterio), non sono solitamente guidati da una definizione formale di approssimazione. Nel paper citato, gli autori forniscono un algoritmo evolutivo utilizzabile con una definizione formale di approssimazione, per problemi multi obiettivo.

Tutti i risultati in letteratura citati finora approssimano il problema della ricerca di punti Pareto efficienti, per problemi di ottimizzazione multi obiettivo, avendo come obiettivo lo studio della complessità temporale di algoritmi per la ricerca di soluzioni approssimate. Avvicinandosi alla prospettiva della Tdg, in particolare quella cooperativa, in [13] viene dimostrato che il problema di razionalizzare l'utilità all'interno di una coalizione, in modo che risulti essere Pareto efficiente, è un problema  $\mathcal{NP}$ -completo se la funzione di scelta seleziona tutte le soluzioni Pareto efficienti, e risulta ancora  $\mathcal{NP}$ -completo se la funzione di scelta contiene una o più soluzioni Pareto efficienti e sono disponibili informazioni sulle preferenze singole degli individui. In [3] gli autori si focalizzano sulla classe di giochi (cooperativa) *hedonic games* e forniscono un algoritmo che calcola soluzioni individualmente razionali e Pareto ottimali per questa categoria di giochi. In generale per questo tipo di giochi, il problema risulta intrattabile.

Mentre è noto ([15]) che la verifica di una soluzione Pareto efficiente in un gioco bimatrice sia facile, non sono noti risultati per il problema di ricerca di soluzioni Pareto efficienti in giochi non cooperativi.

## Capitolo 2

# La curva di Pareto nei giochi $2 \times 2$

### 2.1 L'insieme $Imm(\mathbf{u})$

#### 2.1.1 Preambolo

Per poter calcolare e classificare la Curva di Pareto (forte) in un gioco a 2 giocatori in forma normale, vogliamo inizialmente fornire una classificazione dell'insieme immagine delle funzioni di utilità dei giocatori.

Un gioco  $2 \times 2$  in forma strategica è definito da  $(N, A, V)$ , dove:

$$\begin{aligned} N &= \{1, 2\} \\ A &= \{\{a_1, a_2\}, \{a_3, a_4\}\} \\ V &= \{u_1, u_2\} \end{aligned}$$

Rappresentato comunemente da una *bimatrice*:

$$U = \begin{pmatrix} (A_1, A_2) & (B_1, B_2) \\ (C_1, C_2) & (D_1, D_2) \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \quad (2.1)$$

Definiamo quindi il vettore utilità dei giocatori  $\mathbf{u}(\mathbf{x}) = (u_1(x, y), u_2(x, y))$ :

$$\mathbf{u} : \Delta_1 \times \Delta_2 \rightarrow \mathbb{R}^2$$

$$\begin{cases} u_1(x, y) = x \cdot (y \cdot A_1 + (1 - y) \cdot B_1) + (1 - x) \cdot (y \cdot C_1 + (1 - y) \cdot D_1) \\ u_2(x, y) = x \cdot (y \cdot A_2 + (1 - y) \cdot B_2) + (1 - x) \cdot (y \cdot C_2 + (1 - y) \cdot D_2) \end{cases} \quad (2.2)$$

Dove  $\Delta_1$  e  $\Delta_2$  rappresentano i semplici in cui i giocatori possono selezionare la propria strategia, cioè  $0 \leq x \leq 1$  e  $0 \leq y \leq 1$ .

Il punto di partenza della nostra analisi è l'insieme  $Imm(\mathbf{u})$ :

$$Imm(\mathbf{u}) := \{(u_1, u_2) \mid (u_1, u_2) = \mathbf{u}(\mathbf{x}_1, \mathbf{x}_2) \forall \mathbf{x}_1 \in \Delta_1, \forall \mathbf{x}_2 \in \Delta_2\}$$

Esso rappresenta l'insieme di tutti i payoffs effettivamente riscuotibili dai giocatori nel gioco considerato. A patto di scegliere  $A, B, C, D$  limitati, è evidente che tale insieme risulta compatto e semplicemente connesso.

Inoltre sia  $CO(Z)$  l'involucro convesso di un generico insieme  $Z$  si ha

$$Imm(\mathbf{u}) \subseteq CO(\{A, B, C, D\}) \quad (2.3)$$

Infatti  $CO(\{A, B, C, D\})$  rappresenta l'insieme dei payoff ottenibili attraverso una qualsiasi combinazione convessa tra gli elementi di  $U$ , cioè assegnando una probabilità marginale ad ognuno di essi. Questo insieme è estremamente rilevante perchè la ricerca di equilibri correlati [2], avviene in esso. Abbiamo quindi un sovrainsieme di  $Imm(\mathbf{u})$  di facile descrizione: nel piano  $(u_1, u_2)$  esso risulta un poligono convesso.

Lo studio di  $Imm(\mathbf{u})$  e di un suo sovrainsieme convesso come passo iniziale della ricerca della Curva di Pareto, che indicheremo con  $P(U)$  è naturale poichè tale curva deve necessariamente essere posizionata sulla frontiera di  $Imm(\mathbf{u})$ , indicata con  $\partial Imm(\mathbf{u})$ .

Definiamo ora degli insiemi che in seguito risulteranno comodi nella trattazione. Siano:

$$S = \{AB, AC, BD, CD\}$$

$$P = \{A, B, C, D\}$$

$S$  è l'insieme dei segmenti nello spazio di utilità corrispondenti ai profili di strategie in cui uno dei giocatori utilizza una strategia pura.

$P$  è l'insieme dei punti nello spazio di utilità corrispondenti ai profili di strategie in cui entrambi i giocatori utilizzano una strategia pura.

Prima di illustrare alcune proprietà di  $\partial Imm(\mathbf{u})$  è necessario fare un richiamo di geometria differenziale.

### 2.1.2 Involuppo di una famiglia di curve

Data una famiglia di curve con variabile indipendente  $t$ , dipendenti da un parametro  $c$  nella forma:

$$\begin{cases} x = f(t, c) \\ y = g(t, c) \end{cases} \quad (2.4)$$

**Definizione** L'involuppo è una curva che tocca *ogni* membro della famiglia *tangenzialmente* [19, 36]. L'equazione di tale curva è individuata dal seguente sistema

$$\begin{cases} x = f(t, c) \\ y = g(t, c) \\ 0 = \frac{\partial f}{\partial t} \cdot \frac{\partial g}{\partial c} - \frac{\partial f}{\partial c} \cdot \frac{\partial g}{\partial t} \end{cases} \quad (2.5)$$

Il sistema (2.5) in generale può non ammettere soluzione.

Queste definizioni, ed in particolare le soluzioni del sistema (2.5), verranno sfruttate per fornire un semplice metodo geometrico con il quale è possibile generare  $Imm(\mathbf{u})$ . Infatti le funzioni di utilità  $u_1$  e  $u_2$  di (2.2) dipendono esclusivamente dalle due variabili  $x$  e  $y$ . Allontanandoci per un attimo dalla rappresentazione del gioco (2.1) possiamo immaginare che una della due variabili, diciamo  $y$ , smetta di essere una variabile contestuale e che sia invece un parametro del meccanismo di gioco stesso. In altre parole vogliamo considerare un gioco degenerare con un solo giocatore (con 2 azioni), i cui payoff sono parametrizzati rispetto ad  $y$ . Se questa interpretazione stravolge il meccanismo di gioco originario, non altera però in nessun modo l'insieme immagine delle funzioni di utilità dei giocatori. Abbiamo così una famiglia di curve (segmenti) di utilità che variano con il parametro  $y$ . Possiamo quindi riscrivere il sistema (2.5) nella forma:

$$\begin{cases} u_1(x, y) = x \cdot (y \cdot A_1 + (1 - y) \cdot B_1) + (1 - x) \cdot (y \cdot C_1 + (1 - y) \cdot D_1) \\ u_2(x, y) = x \cdot (y \cdot A_2 + (1 - y) \cdot B_2) + (1 - x) \cdot (y \cdot C_2 + (1 - y) \cdot D_2) \\ 0 = \frac{\partial u_1}{\partial x} \cdot \frac{\partial u_2}{\partial y} - \frac{\partial u_1}{\partial y} \cdot \frac{\partial u_2}{\partial x} \end{cases} \quad (2.6)$$

Dove  $x \in \Delta_1, y \in \Delta_2$ . Vogliamo ora dimostrare qualche proprietà riguardanti le soluzioni del sistema 2.6. Calcolando le derivate parziali, la terza equazione diventa:

$$\begin{aligned} & (y \cdot (A_1 - C_1) + (1 - y) \cdot (B_1 - D_1)) \cdot (x \cdot (A_2 - B_2) + (1 - x) \cdot (C_2 - D_2)) \\ & - (x \cdot (A_1 - B_1) + (1 - x) \cdot (C_1 - D_1)) \cdot (y \cdot (A_2 - C_2) + (1 - y) \cdot (B_2 - D_2)) = 0 \end{aligned}$$

Svolgendo i prodotti semplificando, e assumendo:

$$A_2C_1 - A_1C_2 + B_1C_2 - B_2C_1 + A_1D_2 - A_2D_1 + B_2D_1 - B_1D_2 \neq 0 \quad (2.7)$$

Otteniamo:

$$y = \frac{(B_1C_2 - B_2C_1 + B_2D_1 - B_1D_2 + C_1D_2 - C_2D_1)}{A_2C_1 - A_1C_2 + B_1C_2 - B_2C_1 + A_1D_2 - A_2D_1 + B_2D_1 - B_1D_2} + \frac{x \cdot (A_2B_1 - A_1B_2 + B_2C_1 - B_1C_2 + A_1D_2 - A_2D_1 + C_2D_1 - C_1D_2)}{A_2C_1 - A_1C_2 + B_1C_2 - B_2C_1 + A_1D_2 - A_2D_1 + B_2D_1 - B_1D_2}$$

Che, facendo dei raccoglimenti, può essere espressa anche come:

$$y = \frac{((A_1 - C_1)(B_2 - D_2) - (A_2 - C_2)(B_1 - D_1)) \cdot x}{(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1)} - \frac{(B_1C_2 - B_2C_1 + B_2D_1 - B_1D_2 + C_1D_2 - C_2D_1)}{(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1)}$$

Poniamo:

$$M = \frac{(A_1 - C_1)(B_2 - D_2) - (A_2 - C_2)(B_1 - D_1)}{(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1)}$$

$$K = -\frac{(B_1C_2 - B_2C_1 + B_2D_1 - B_1D_2 + C_1D_2 - C_2D_1)}{(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1)}$$

Da cui:

$$y = M \cdot x + K \quad (2.8)$$

Vale quindi:

**Teorema** Le soluzioni del sistema (2.6), se esistono, stabiliscono una relazione lineare tra  $x$  ed  $y$ .

Sostituendo l'equazione (2.8) in  $u_1(x, y)$  e  $u_2(x, y)$  otteniamo:

$$\begin{cases} u_1(x) = (B_1 - D_1 + K(A_1 - B_1 + D_1 - C_1) + M(C_1 - D_1)) \cdot x \\ \quad + M(A_1 - B_1 + D_1 - C_1) \cdot x^2 + D_1 + K(C_1 - D_1) \\ u_2(x) = (B_2 - D_2 + K(A_2 - B_2 + D_2 - C_2) + M(C_2 - D_2)) \cdot x \\ \quad + M(A_2 - B_2 + D_2 - C_2) \cdot x^2 + D_2 + K(C_2 - D_2) \end{cases} \quad (2.9)$$

Per cui:

**Teorema** Le funzioni di utilità  $u_1$  e  $u_2$  parametrizzate rispetto alle soluzioni dell'involuppo, sono funzioni quadratiche in una sola variabile.

Verrà ora enunciato un teorema generale di geometria, che verrà usato come punto di partenza per trarre delle conclusioni importanti sul sistema (2.6) :

**Teorema** Una curva piana in forma parametrica polinomiale di grado 2 nella forma:

$$\begin{cases} u_1 = f(t) = R_1t^2 + R_2t + R_3 \\ u_2 = g(t) = R_4t^2 + R_5t + R_6 \end{cases} \quad (2.10)$$

Con  $t \in \mathbb{R}$ , è una parabola.

**Dimostrazione** Assumiamo  $R_1 \neq 0$  e  $R_4 \neq 0$ . Se così non fosse, il sistema (2.10) descriverebbe evidentemente una retta nel piano  $(u_1, u_2)$ . Riscriviamo il sistema (2.10) come:

$$\begin{cases} \frac{u_1}{R_1} = t^2 + \frac{R_2}{R_1}t + \frac{R_3}{R_1} \\ \frac{u_2}{R_4} = t^2 + \frac{R_5}{R_4}t + \frac{R_6}{R_4} \end{cases}$$

Sottraiamo la seconda equazione dalla prima facendo sparire il termine di secondo grado e risolviamo rispetto a  $t$ :

$$t = \frac{R_4(u_1 - R_3) - R_1(u_2 - R_6)}{R_2R_4 - R_1R_5}$$

Infine sostituiamo  $t$  nella prima equazione del sistema (2.10) :

$$u_1R_1 \left( \frac{R_4(u_1 - R_3) - R_1(u_2 - R_6)}{R_2R_4 - R_1R_5} \right)^2 + R_2 \left( \frac{R_4(u_1 - R_3) - R_1(u_2 - R_6)}{R_2R_4 - R_1R_5} \right) + R_3$$

Eliminando i denominatori e raccogliendo possiamo ad arrivare alla forma:

$$Eu_2^2 + Fu_1u_2 + Gu_2^2 + Hu_1 + Iu_2 + L = 0$$

In cui

$$\begin{aligned} E &= R_1R_4^2 \\ F &= -2R_1^2R_4 \\ G &= R_1^3 \end{aligned}$$

Poichè  $F^2 = 4EG$ , la curva è una parabola, come volevasi dimostrare. Nel caso  $R_2R_4 - R_1R_5 = 0$  (come anche nel caso  $R_1 = R_4 = 0$ ) la parabola è degenera, e rappresenta una retta.

Infine enunciamo il teorema conclusivo di questa sottosezione, che riguarda direttamente il sistema 2.6:

**Teorema** Il sistema (2.6) ammette come soluzione nel piano (se esiste) un tratto di parabola *non* degenerare, o un punto.

**Dimostrazione** Poniamoci nel caso in cui valga l'equazione (2.7), che può essere scritta anche come:

$$(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1) \neq 0 \quad (2.11)$$

Con questa ipotesi, possiamo derivare dal sistema (2.6) le equazioni (2.9) che rappresentano nel piano  $(u_1, u_2)$  le soluzioni (se esistono) del sistema. Come corollario del teorema precedente sappiamo che sicuramente queste costituiranno una parabola. Dobbiamo dimostrare che se la parabola degenera in una retta, allora esiste *al più* un punto nel piano  $(u_1, u_2)$  che è soluzione del sistema (2.6). La degenerazione in una retta avviene se:

$$(A_1M - B_1M - C_1M + D_1M) = (A_2M - B_2M - C_2M + D_2M) = 0$$

Oppure se

$$\frac{M(A_1 - B_1 + D_1 - C_1)}{M(A_2 - B_2 + D_2 - C_2)} = \frac{B_1 - D_1 + K(A_1 - B_1 - C_1 + D_1) + M(C_1 - D_1)}{B_2 - D_2 + K(A_2 - B_2 + D_2 - C_2) + M(C_2 - D_2)} \quad (2.12)$$

Cominciamo dal primo caso.

Raccogliendo  $M$  abbiamo:

$$M(A_1 - B_1 + D_1 - C_1) = M(A_2 - B_2 + D_2 - C_2) = 0$$

Se  $M = 0$ , allora:

$$(A_1 - C_1)(B_2 - D_2) - (A_2 - C_2)(B_1 - D_1) = 0 \quad (2.13)$$

e

$$y = K$$

Esplicitando  $A_1$  rispetto alle altre costanti nell'equazione (2.13) e sostituendo

il valore di  $y$  nelle funzioni di utilità dei giocatori, il sistema (2.6) diventa:

$$\begin{cases} A_1 = \frac{A_2 B_1 + B_2 C_1 - B_1 C_2 - A_2 D_1 + C_2 D_1 - C_1 D_2}{B_2 - D_2} \\ u_1(x, K) = x \cdot (K \cdot A_1 + (1 - K) \cdot B_1) + (1 - x) \cdot (K \cdot C_1 + (1 - K) \cdot D_1) \\ u_2(x, K) = x \cdot (K \cdot A_2 + (1 - K) \cdot B_2) + (1 - x) \cdot (K \cdot C_2 + (1 - K) \cdot D_2) \end{cases}$$

In cui sostituendo i valori e semplificando si ottiene:

$$\begin{cases} u_1 = \frac{-B_2 C_1 + A_2 D_1 - C_2 D_1 + C_1 D_2}{A_2 - B_2 - C_2 + D_2} \\ u_2 = \frac{-B_2 C_2 + A_2 D_2}{A_2 - B_2 - C_2 + D_2} \end{cases}$$

Cioè un punto costante nel piano. Viceversa se  $M \neq 0$  allora

$$(A_1 - B_1 + D_1 - C_1) = (A_2 - B_2 + D_2 - C_2) = 0$$

Da cui otteniamo:

$$(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1) = 0$$

Che è contro l'ipotesi (2.11). Rimane da analizzare il secondo caso che può dar luogo ad una degenerazione della parabola.

Se vale l'equazione (2.12) possiamo assumere  $M \neq 0$  (se  $M$  fosse nullo si ricadrebbe nel caso precedente) e ottenere quindi:

$$\frac{(A_1 - B_1 + D_1 - C_1)}{(A_2 - B_2 + D_2 - C_2)} = \frac{B_1 - D_1 + K(A_1 - B_1 + D_1 - C_1) + M(C_1 - D_1)}{B_2 - D_2 + K(A_2 - B_2 + D_2 - C_2) + M(C_2 - D_2)}$$

Svolgendo i prodotti e semplificando arriviamo a:

$$A_2 C_1 - A_1 C_2 + B_1 C_2 - B_2 C_1 + A_1 D_2 - A_2 D_1 + B_2 D_1 - B_1 D_2 = 0$$

Che è ancora contro l'ipotesi (2.13) (equivalente a (2.11)).

Abbiamo quindi dimostrato che, nell'ipotesi (2.11) il teorema è verificato. Se questa ipotesi non dovesse valere, non sarebbe possibile esprimere  $y$  in funzione di  $x$ , ma varrebbe:

$$x = K'$$

Con  $K'$  costante. A quel punto, esprimendo il sistema (2.9) in funzione di  $y$  e ripetendo la dimostrazione appena svolta con i coefficienti opportuni, si arriverebbe ad analoghe conclusioni. Il teorema è quindi dimostrato.

### 2.1.3 Proprietà della frontiera $\partial Imm(\mathbf{u})$

**Teorema** In generale  $\partial Imm(\mathbf{u})$  è definita a tratti da segmenti ed *eventualmente* da un tratto di curva non lineare (parabola). In particolare, se  $r \in \partial Imm(\mathbf{u})$  è un segmento, allora:

$$\exists s \in S, r \subseteq s$$

Se invece  $c \in \partial Imm(\mathbf{u})$  è una curva non lineare, allora tutti i suoi punti sono esattamente le soluzioni di (2.6).

**Dimostrazione** Riprendiamo l'interpretazione geometrica dell'insieme  $Imm(\mathbf{u})$  fornita precedentemente: supponiamo che esso sia generato da una famiglia di segmenti che varino parametricamente rispetto alla strategia di uno dei giocatori, diciamo  $y$ . Abbiamo quindi che per  $y = 0$  e  $y = 1$ , cioè gli estremi dell'intervallo su cui è definito il parametro  $y$ , i segmenti ottenuti sono rispettivamente  $BD$  e  $AC$ . Se avessimo eseguito lo stesso ragionamento utilizzando come parametro  $x$ , avremmo ottenuto che i segmenti generati per valori del parametro agli estremi del suo intervallo di definizione, sarebbero stati  $AB$  e  $CD$ . Questo non stupisce: stiamo di fatto considerando tutti i profili di strategie in cui un giocatore gioca una strategia pura, ed un altro una mista. In altre parole stiamo considerando righe e colonne della bimatrice di partenza. Per convincerci ulteriormente di quanto detto possiamo ricordare l'equazione (2.3): fatte salve le combinazioni convesse tra elementi delle diagonali della bimatrice,  $\partial CO(\{A, B, C, D\})$  è costituita unicamente da elementi di  $S$ .

Resta da dimostrare la parte del teorema relativa al tratto di curva non lineare. Inizialmente mostreremo che se esiste una curva non lineare in  $\partial Imm(\mathbf{u})$  allora soddisfa (2.6) e successivamente che, se esiste, tale curva è unica.

Ricordiamo una condizione necessaria affinché un punto  $(u_1, u_2)$  sia Pareto efficiente (una versione delle condizioni KKT, ([6]):

**Proprietà** Se  $(\bar{u}_1, \bar{u}_2)$  è una soluzione Pareto efficiente del gioco considerato, allora  $\exists \lambda_1, \lambda_2$  non entrambi nulli tali per cui  $(\bar{u}_1, \bar{u}_2)$  massimizza:

$$f = \lambda_1 u_1 + \lambda_2 u_2$$

Abbiamo quindi una condizione necessaria su  $(\bar{u}_1, \bar{u}_2)$  per la proprietà Pareto efficienza. In corrispondenza di un massimo di  $f$  abbiamo:

$$\nabla f = 0$$

Cioè:

$$\begin{cases} \lambda_1 u_{1x} + \lambda_2 u_{2x} = 0 \\ \lambda_1 u_{1y} + \lambda_2 u_{2y} = 0 \end{cases}$$

Spostiamo dei termini a destra:

$$\begin{cases} \lambda_1 u_{1x} = -\lambda_2 u_{2x} \\ \lambda_2 u_{2y} = -\lambda_1 u_{1y} \end{cases}$$

Moltiplichiamo ora la prima equazione per la seconda, ottenendo:

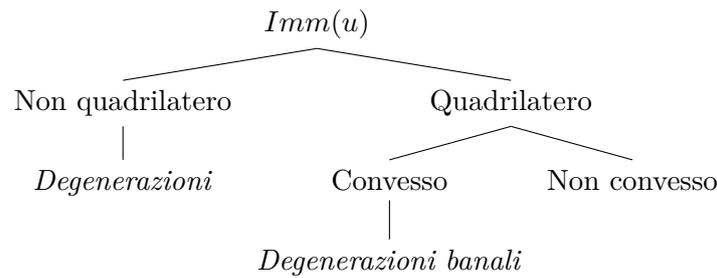
$$\lambda_1 \lambda_2 u_{1x} u_{2y} = \lambda_1 \lambda_2 u_{1y} u_{2x}$$

Semplificando:

$$u_{1x} u_{2y} - u_{1y} u_{2x} = 0$$

Che è esattamente la terza equazione del sistema (2.6).

Abbiamo quindi dimostrato che una qualsiasi curva non lineare Pareto efficiente, è una soluzione del sistema (2.6). Inoltre per la definizione di inviluppo fornita precedentemente, se esiste tale curva, è unica. È lecito domandarsi se questo risultato sia esattamente ciò che volevamo ottenere: sicuramente una curva Pareto efficiente appartiene a  $\partial Imm(\mathbf{u})$ , ma non possiamo escludere che ci siano altri tratti di curva appartenenti a  $\partial Imm(\mathbf{u})$  che non siano Pareto efficienti, nemmeno a seguito di rotazioni arbitrarie su  $Imm(\mathbf{u})$ . In realtà sarà chiaro in seguito che non è questo il caso: se esistono tratti di curva appartenenti a  $\partial Imm(\mathbf{u})$  che non sono Pareto efficienti, essi sono tratti della curva che è soluzione di (2.6).

Figura 2.1: Albero di classificazione di  $Imm(\mathbf{u})$ .

#### 2.1.4 Classificazione di $Imm(\mathbf{u})$

Vogliamo ora fornire una classificazione efficace dei giochi bimatrice, basata sulle proprietà geometriche del relativo insieme  $Imm(\mathbf{u})$ . Un gioco bimatrice è completamente determinato dalla bimatrice (2.1)  $U$ , cioè in generale da 8 parametri. Un primo passo nello studio di  $\mathbf{u}$  potrebbe essere quello di ricordare che la proprietà di Pareto efficienza (quindi la relazione di Pareto dominanza) ed in generale le proprietà geometriche di  $Imm(\mathbf{u})$ , sono invarianti a trasformazioni affini di  $u_1$  e  $u_2$  (anche differenti). Possiamo quindi ridurre il numero di parametri a 6, riconducendoci sempre al caso  $A = (A_1, A_2) = (0, 0)$ . Non riuscendo a ridurre ulteriormente il numero di parametri, eviteremo un approccio completamente analitico, che risulterebbe particolarmente pesante. Senza perdita di precisione e rigore, ci affideremo quindi al buon senso e all'intuizione geometrica, che ci guideranno nel nostro percorso e che verranno confermati dai risultati analitici precedentemente ottenuti. Tutta l'analisi verrà effettuata avendo come riferimento l'interpretazione di  $Imm(\mathbf{u})$  fornita nella Sezione (2.1.2).

Consideriamo quindi il caso più generale possibile:  $A \neq B \neq C \neq D$  non allineati tra loro, nel seguito si analizzeranno anche i casi particolari e degeneri.<sup>1</sup>

**Teorema** Le proprietà dell'insieme  $S$  determinano completamente  $Imm(\mathbf{u})$ .

**Dimostrazione** Siano:

$$r, s \in S$$

$$p, q \in P$$

In generale possiamo distinguere 2 casi:

1.  $S$  determina un quadrilatero nel piano  $(u_1, u_2)$ . In altre parole nessun elemento di  $S$  ha punti comuni con gli altri, fatti salvi quelli banali (gli estremi). Dal punto di vista geometrico i segmenti sono consecutivi a 2 a 2. In formule:

<sup>1</sup>Le formule utilizzate per differenziare i due casi principali, sono valide sotto la condizione che non vi siano punti allineati. La collocazione delle degenerazioni in una casistica piuttosto che in un'altra, sarà chiara in seguito.

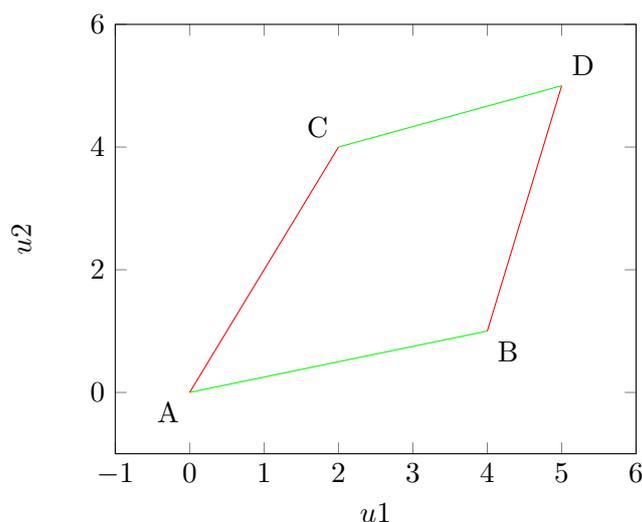


Figura 2.2:  $Imm(\mathbf{u})$  Quadrilatero convesso.

$$\forall r \forall s \forall p$$

$$(r \cap s \neq p) \Rightarrow (r \cap s = \emptyset \vee \exists q (r \cap s = q)) \quad (2.14)$$

A sua volta il quadrilatero può essere :

- (a) Convesso.
  - (b) Non convesso,
2.  $S$  non determina un quadrilatero nel piano  $(u_1, u_2)$ , cioè esiste una coppia di segmenti di  $S$  che s'interseca in un punto non banale e (a meno di degenerazioni) è unica. In formule:

$$\exists r \exists s$$

$$(r \cap s \neq \emptyset \wedge \forall p (r \cap s \neq p)) \quad (2.15)$$

**Caso 1: Quadrilatero** Nell'eventualità 1a di quadrilatero convesso, il sistema (2.6) non ammette soluzione. Non esiste quindi nessun tratto di involuppo che appartenga a  $\partial Imm(\mathbf{u})$ .

**Esempio:** La seguente bimatrice rappresenta un gioco in cui  $Imm(\mathbf{u})$  è un quadrilatero convesso. Tale insieme è rappresentato nella Figura (2.2).

$$\begin{pmatrix} (0,0) & (4,1) \\ (2,4) & (5,5) \end{pmatrix} \quad (2.16)$$

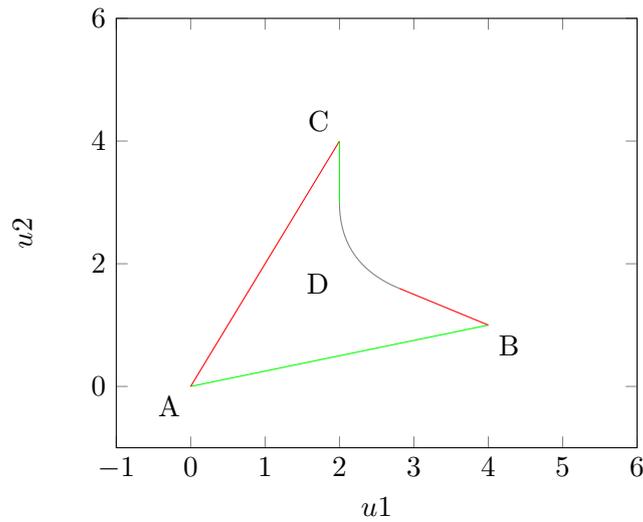


Figura 2.3:  $Imm(\mathbf{u})$  quadrilatero non convesso.

Corollario Nel caso 1a ogni possibile outcome è all'interno del quadrilatero, determinando così

$$\partial Imm(\mathbf{u}) := S$$

Nel caso di quadrilatero non convesso invece si vuole provare la seguente proprietà.

**Proprietà** Nel caso 1b il sistema (2.6) ammette sempre soluzioni. In particolare le soluzioni di questo sistema sono localizzate all'esterno del quadrilatero, nella regione compresa tra i due lati che formano l'angolo non convesso. Tutte le soluzioni del sistema appartengono a  $\partial Imm(\mathbf{u})$ .

**Dimostrazione** Consideriamo senza perdita di generalità il caso della Figura 2.3. Sia  $dir(s)$  la retta direttrice del segmento  $s$ . Allora:

$$\exists \bar{x}, 0 < \bar{x} < 1 \mid (AC \cap dir(BD)) = \bar{x} \cdot A + (1 - \bar{x}) \cdot C$$

Il punto di tangenza dell'involuppo sul lato  $BD$  risulta quindi:

$$\bar{x} \cdot B + (1 - \bar{x}) \cdot D$$

Analogamente:

$$\exists \bar{y}, 0 < \bar{y} < 1 \mid (CD \cap dir(AB)) = \bar{y} \cdot C + (1 - \bar{y}) \cdot D$$

Il punto di tangenza dell'involuppo sul lato  $AB$  risulta quindi:

$$\bar{y} \cdot A + (1 - \bar{y}) \cdot B$$

La conoscenza di  $\bar{y}$  e  $\bar{x}$ , calcolabili in tempo costante, ci permette di avere dei constraints su quelle che saranno le soluzioni del sistema (2.6). In particolare, siano  $(x_{inv}, y_{inv})$  tutte e sole le soluzioni del sistema, allora è possibile esprimere i seguenti vincoli in funzione del vertice dell'angolo non convesso  $E$ :

$$\begin{cases} \bar{x} \leq x_{inv} \leq 1 \\ \bar{y} \leq y_{inv} \leq 1 \end{cases} \quad se \quad E = A$$

$$\begin{cases} \bar{x} \leq x_{inv} \leq 1 \\ 0 \leq y_{inv} \leq \bar{y} \end{cases} \quad se \quad E = B$$

$$\begin{cases} 0 \leq x_{inv} \leq \bar{x} \\ \bar{y} \leq y_{inv} \leq 1 \end{cases} \quad se \quad E = C$$

$$\begin{cases} 0 \leq x_{inv} \leq \bar{x} \\ 0 \leq y_{inv} \leq \bar{y} \end{cases} \quad se \quad E = D$$

**Esempio:** La seguente bimatrice rappresenta un gioco in cui  $Imm(\mathbf{u})$  è un quadrilatero *non* convesso. Tale insieme è rappresentato nella Figura 2.3.

$$\begin{pmatrix} (0, 0) & (4, 1) \\ (2, 4) & (2, 2) \end{pmatrix} \quad (2.17)$$

**Caso 2: non quadrilatero** Facendo riferimento a (2.15), senza perdita di generalità consideriamo il caso  $r = AB$  e  $s = CD$ . Il punto di intersezione  $K = r \cap s$  può essere espresso come:

$$\begin{cases} K = \bar{y} \cdot A + (1 - \bar{y}) \cdot B \\ K = \hat{y} \cdot C + (1 - \hat{y}) \cdot D \end{cases} \quad (2.18)$$

**Proprietà** Facendo riferimento all'interpretazione geometrica di  $Imm(\mathbf{u})$  fornita in Sezione 2.1.2, risulta evidente che esistono soluzioni distinte al sistema

(2.6) se e solo se  $\hat{y} \neq \bar{y}$ , e queste soluzioni costituiscono un insieme di misura non nulla. Se invece  $\hat{y} = \bar{y}$  il sistema ammette un'unica soluzione  $K = (u_1(x, \bar{y}), u_2(x, \bar{y})) \forall x$ . Questo è stato dimostrato dall'analisi svolta sulle proprietà del sistema (2.6) nelle sottosezioni precedenti. In particolare il caso in cui l'involuppo dava luogo ad una soluzione nel piano  $(u_1, u_2)$  costante era associato alle condizioni:

$$(A_1 - C_1)(B_2 - D_2) - (A_2 - C_2)(B_1 - D_1) = 0$$

Oppure:

$$(A_1 - B_1)(C_2 - D_2) - (A_2 - B_2)(C_1 - D_1) = 0$$

Che possono essere riscritte come:

$$\frac{A_2 - C_2}{A_1 - C_1} = \frac{B_2 - D_2}{B_1 - D_1} \quad (2.19)$$

e

$$\frac{A_2 - B_2}{A_1 - B_1} = \frac{C_2 - D_2}{C_1 - D_1} \quad (2.20)$$

Che rappresentano rispettivamente la condizione di parallelismo tra  $AC, BD$  e  $AB, CD$ <sup>2</sup>. È evidente che affermare che  $\hat{y} = \bar{y}$  equivale a dire che vale la condizione (2.19) ma non (2.20). È importante notare anche che affinché l'involuppo di misura nulla esista, le condizioni (2.19) e (2.20) sono in un OR mutuamente esclusivo. Infatti se tutti gli elementi di  $S$  fossero paralleli a 2 a 2 otterremmo come  $Imm(\mathbf{u})$  un parallelogramma, cioè un quadrilatero (ed infatti la terza equazione del sistema (2.6) non darebbe alcuna informazione su  $x$  e  $y$ , rendendo in generale impossibile il sistema). Ad ogni modo, sia che  $\hat{y} \neq \bar{y}$  o  $\hat{y} = \bar{y}$ , le soluzioni del sistema (2.6) appartengono a  $\partial Imm(\mathbf{u})$ .

Nel caso  $\hat{y} \neq \bar{y}$  sia  $\hat{y} < \bar{y}$  allora è possibile esprimere un constraint su  $y_{inv}$ :

$$\hat{y} \leq y_{inv} \leq \bar{y}$$

**Esempio:** La seguente bimatrice rappresenta un gioco in cui  $Imm(\mathbf{u})$  non è un quadrilatero, ed in cui esiste un tratto di involuppo di misura non nulla appartenente a  $\partial Imm(\mathbf{u})$ .  $Imm(\mathbf{u})$  è rappresentato nella Figura (2.4).

$$\begin{pmatrix} (0, 0) & (4, 2) \\ (1, 5) & (2, 0) \end{pmatrix} \quad (2.21)$$

<sup>2</sup>Stiamo implicitamente assumendo che i denominatori siano diversi da zero. Questo non lede la generalità del ragionamento in quanto avendo supposto punti non allineati (e quindi neanche identici) tra loro, se il denominatore a sinistra dell'uguale fosse nullo, lo sarebbe anche quello a destra, confermando la tesi di parallelismo.

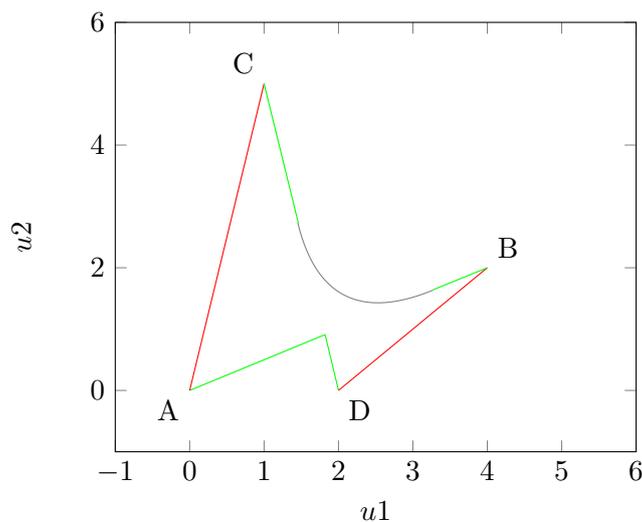


Figura 2.4:  $Imm(\mathbf{u})$  Non quadrilatero.

**Casi degeneri** Una prima forma di degenerazione è stata citata illustrando il caso non quadrilatero: facendo riferimento a (2.18), se  $\hat{y} = \bar{y}$  il sistema (2.6) avrà un'unica soluzione di misura nulla. In altre parole l'involuppo appartenente a  $\partial Imm(\mathbf{u})$  crollerà su un unico punto. Chiameremo questo caso *non quadrilaterale lineare a tratti*.

La prossima bimatrice, il cui insieme immagine delle utilità è rappresentato in Figura 2.5 descrive proprio questa situazione:

$$\begin{pmatrix} (0,0) & (5,0) \\ (5,4) & (0,4) \end{pmatrix} \quad (2.22)$$

Altre 2 degenerazioni particolarmente rilevanti rientrano ancora nella casistica di  $Imm(\mathbf{u})$  non quadrilaterale.

Una prima degenerazione appare quando, nel sistema (2.18),  $\hat{y} \neq \bar{y}$  ma il punto  $K$  coincide con elemento dell'insieme  $P$ . Questo determina la sparizione di una regione «triangolare» all'interno di  $Imm(\mathbf{u})$  altrimenti presente ed il fatto che, a differenza di tutti i casi illustrati precedentemente, l'involuppo risulta essere tangente ad un elemento  $s \in S$  in corrispondenza di un elemento  $p \in P$ , cioè di un vettore di payoffs puro. Esempio:

$$\begin{pmatrix} (0,0) & (4,2) \\ (1,5) & (2,1) \end{pmatrix} \quad (2.23)$$

Il grafico è illustrato in Figura 2.6.

La seconda degenerazione più rilevante si verifica se e solo se i payoffs di una (e una sola) delle diagonali della bimatrice sono uguali, i.e.  $A = D$  o  $B = C$ . Il siste-

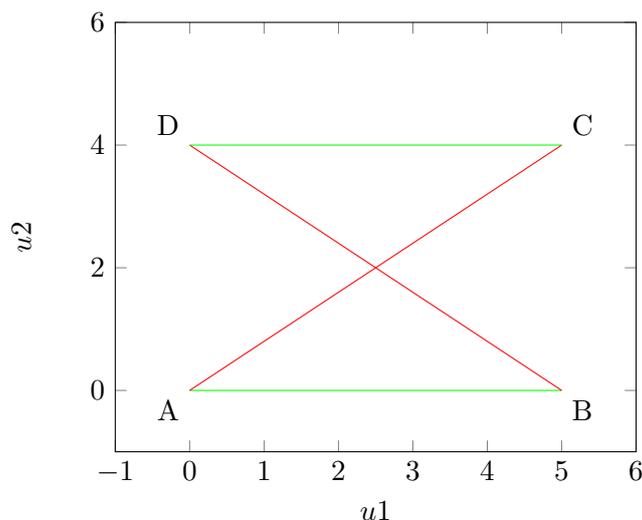


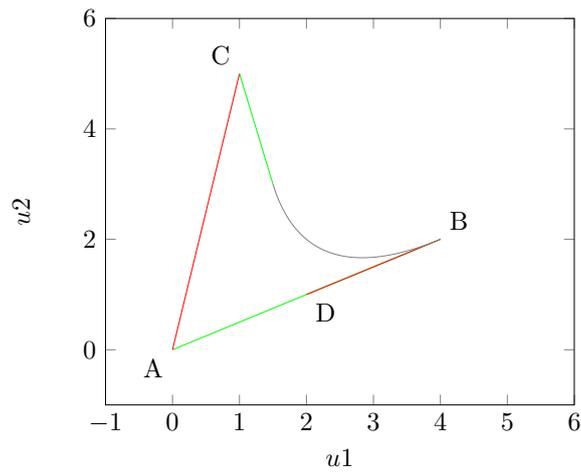
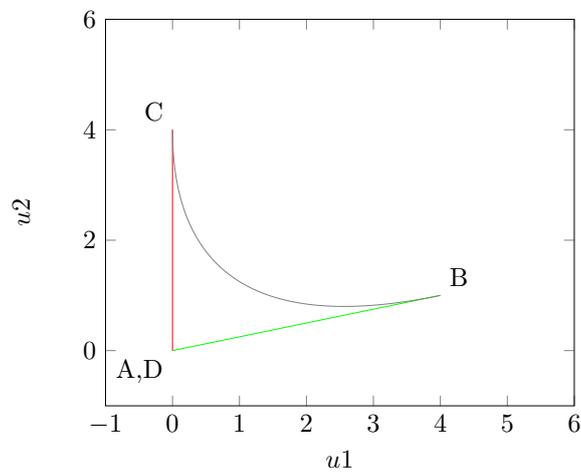
Figura 2.5:  $Imm(\mathbf{u})$  Non quadrilatero lineare a tratti.

ma (2.6) ammette come soluzione una curva che congiunge i punti della rimanente diagonale della bimatrice  $BC$  o  $AD$ . Tale curva appartiene a  $\partial Imm(\mathbf{u})$ . Esempio:

$$\begin{pmatrix} (0,0) & (4,1) \\ (0,4) & (0,0) \end{pmatrix} \quad (2.24)$$

Il grafico è illustrato in Figura 2.7.

Per completare la classificazione di tutte le possibili forme di  $Imm(\mathbf{u})$  bisogna citare un altro tipo di degenerazioni possibile riguardante la casistica quadrilaterale convessa. Questo tipo di degenerazioni non presenta importanti novità geometriche, ma si tratta semplicemente di contrazioni del caso generale di quadrilatero convesso. La prima contrazione che possiamo effettuare è la seguente: supponiamo che 2 vettori di payoffs puri, posti sulla stessa riga o colonna, siano uguali. Come si può immaginare  $Imm(\mathbf{u})$  avrà a questo punto forma triangolare. La seconda è ottenibile uguagliando 3 vettori di payoffs puri:  $Imm(\mathbf{u})$  sarà un segmento. Lo stesso forma a segmento è ottenibile alternativamente allineando gli outcome puri tra loro. Infine, sebbene sia un caso del tutto privo di interesse, si può procedere contraendo ulteriormente il segmento, fino a che tutti gli outcome puri siano identici ed  $Imm(\mathbf{u})$  sia costituito da un solo punto.

Figura 2.6:  $Imm(\mathbf{u})$  non quadrilatero degenero 1.Figura 2.7:  $Imm(\mathbf{u})$  non quadrilatero degenero 2.

Poichè per i nostri scopi le degenerazioni del quadrilatero convesso non presentano nessuna variazione concettuale significativa rispetto ad esso, chiameremo queste degenerazioni *semplici*, per distinguerle da quelle illustrate in precedenza. Si lascia al lettore il semplice compito di definire bimatrici che rientrino in queste casistiche. È importante notare che questa è una classificazione esaustiva: partendo da una qualsiasi delle casistiche individuate e variando i parametri a proprio piacimento, si può modificare la topologia di  $Imm(\mathbf{u})$ , ma si finirà sempre in uno dei casi individuati. Riportiamo nella Tabella 2.1 la classificazione dettagliata di tutti gli shapes individuati.

Tabella 2.1: Classificazione dettagliata  $Imm(\mathbf{u})$ .

	Numero di tratti in $\partial Imm(\mathbf{u})$	Inviluppo in $\partial Imm(\mathbf{u})$	Numero di outcome puri in inviluppo
Quadrilatero convesso	4	No	
Quadrilatero conv. degenerare: triangolo	3	No	
Quadrilatero conv. degenerare: segmento	1	No	
Quadrilatero conv. degenerare: punto	1	No	
Quadrilatero non convesso	5	Parabola	0
Non quadrilatero	7	Parabola	0
Non quadrilatero: inviluppo degenerare	4	Punto	0
Non quadrilatero degenerare 1: tre payoffs allineati	4	Parabola	1
Non quadrilatero degenerare 2: due payoffs diagonale uguali	3	Parabola	2

### 2.1.5 Computazione di $\partial Imm(\mathbf{u})$

Nella Sezione 2.1.4 è stata fornita una classificazione esaustiva di tutti i possibili giochi  $2x2$ , rispetto alle proprietà geometriche di  $Imm(\mathbf{u})$ . Ci domandiamo ora quale come sia strutturato un algoritmo che definisca a quale categoria della classificazione fornita nella Tabella 2.1 appartiene, e ne determini completamente  $\partial Imm(\mathbf{u})$ . Per rispondere a questa domanda, riportiamo lo pseudocodice per la classificazione di  $Imm(\mathbf{u})$  nell'Algoritmo 2.1. Le funzioni principali sono le seguenti:

- Funzione  $get2x2GameType(M)$  :  
Determina il tipo di  $Imm(\mathbf{u})$  della bimatrice  $2 \times 2$   $M$  fornita in input: innanzitutto si definiscono gli insiemi  $S$  e  $P$  illustrati nella Sezione 2.1.1. Fatto questo viene richiamata la funzione  $checkDegenerateCases$ : se nessun caso degenero viene individuato, procediamo ricercando i punti di intersezione tra le rette direttrici associate ai segmenti corrispondenti ai profili di strategie in cui lo stesso giocatore gioca una strategia pura<sup>3</sup>. A questo punto è semplice definire le condizioni per discriminare tra quadrilatero convesso, quadrilatero non convesso e non quadrilatero sfruttando la funzione  $isContained()$ .
- Funzione  $checkDegenerateCases(S, P)$  :  
Facendo dei controlli triviali su  $S$  e  $P$  (illustrati nella Sezione 2.1.4) individua tutte le possibili degenerazioni illustrate nella Tabella 2.1 ritornando un valore booleano *salvo il caso non quadrilaterale con involuppo degenero*, che non è identificabile semplicemente ispezionando gli insiemi  $S, P$  ma necessita di calcoli aggiuntivi, che verranno effettuati nella trattazione dei casi non degeneri.
- Funzione  $isContained(p, s)$ :  
Tale funzione restituisce vero se il punto fornito come primo argomento è *strettamente* contenuto nel segmento fornito come secondo argomento, mentre restituisce falso in tutti gli altri casi. È importante notare che le condizioni  $nonQuadcond1$  e  $nonQuadcond2$  sono quindi mutuamente esclusive.
- Funzione  $hasCurveEvenvelope(...)$ :  
Essa calcola  $\bar{y}, \hat{y}$  riportati nelle equazioni (2.18) per valutare se esiste o meno un involuppo non degenero (l'analogo caso in cui valga  $nonQuadCond2$  viene omissso per brevità) e restituisce vero o falso. Sfrutta la mutua esclusione tra le condizioni  $nonQuadcond1$  e  $nonQuadcond2$  per discriminare se c'è intersezione tra i segmenti associati al profilo di strategie in cui il primo giocatore gioca una strategia pura, o tra quelli associati ad uno in cui sia il secondo a farlo.

<sup>3</sup>Non è detto che tali punti esistano. Se ad esempio  $AB$  è parallelo a  $CD$ ,  $k_1$  sarà indefinito.

Nell'Algoritmo 2.2 invece individuato  $\partial Imm(\mathbf{u})$ . Le funzioni principali sono:

- Funzione *computeBorder*( $M$ ):  
Calcola  $\partial Imm(\mathbf{u})$  di una generica bimatrice  $2 \times 2$   $M$ : nel caso di quadrilatero convesso, non quadrilatero con inviluppo degenere ed altri casi degeneri (e.g. triangolo)  $\partial Imm(\mathbf{u})$  è già noto: coincide esattamente con l'insieme  $S$ . Negli altri casi invece sappiamo che ci sarà sicuramente un tratto di parabola (inviluppo) appartenente alla frontiera. Per poter risolvere il sistema (2.6), assumeremo di avere un motore per il calcolo simbolico a darci supporto, che ci permetterà di definire la relazione tra la strategia del primo giocatore e quella del secondo, sul tratto di curva (righe 15, 16 e 33, 34). Poichè  $\partial Imm(\mathbf{u})$  è strutturato in modi diversi, i casi non quadrilatero e quadrilatero vengono gestiti con due funzioni distinte.
- Funzione *computeNonConvexQuadBorder*( $M$ ) e *computeNonQuadBorder*( $M$ ):  
Entrambe le funzioni individuano ad ogni modo i punti estremi dell'inviluppo (*invK1*, *invK2* ed *invK11*, *invK12*) necessari per «tagliare» i segmenti di  $S$  nella definizione di  $\partial Imm(\mathbf{u})$ . Sia nel caso di quadrilatero non convesso, che in quello di non quadrilatero ci sono 4 casi speculari, a seconda di quale sia il vertice associato all'angolo non convesso e tra quale coppia di lati esiste l'inviluppo. Per evitare di essere verbosi viene descritta soltanto una casistica: le altre non aggiungono nessuna informazione alla trattazione. Allo stesso modo omettiamo i 2 casi di non quadrilatero degenere, in quanto gestibili con lievi variazioni rispetto al caso generale. Fatto ciò abbiamo tutte le informazioni necessarie per definire il bordo ricercato: prima si definiranno i tratti lineari appartenenti ad esso<sup>4</sup> (righe 23 e 39), dopodichè si aggiungerà il tratto di curva (righe 24 e 30) utilizzando la funzione *envPiece*(...). .
- Funzione *envPiece*(...):  
Restituisce l'insieme di punti nel piano associati alle funzioni in forma parametrica fornite come primi 2 argomenti, nel range indicato come terzo e quarto argomento.

A questo punto possiamo affermare con sicurezza quanto si era intuito:

**Teorema** La classificazione di  $Imm(\mathbf{u})$  e il calcolo esatto di  $\partial Imm(\mathbf{u})$  in un gioco bimatrice  $2 \times 2$ , hanno una complessità computazionale temporale costante.

---

<sup>4</sup>Un segmento viene denotato utilizzando i nomi dei suoi due punti estremi: ad esempio *CinvK1* (riga 23) indica il segmento tra il punto  $C$  e *invK1*.

**Algoritmo 2.1** Classificazione  $Imm(\mathbf{u})$ 


---

```

1: function GET2X2GAMETYPE(M) ▷ variables used in this function are global
2:   S ← getSegments(M)
3:   P ← getPureOutcomes(M)
4:   GameType ← null
5:   GameType ← checkDegenerateCases(S, P)
6:   if GameType ≠ null then
7:     return GameType
8:   end if
9:   k1 ← solve(dir(AB) = dir(CD))
10:  k2 ← solve(dir(AC) = dir(BD))
11:  nonQuadCond1 ← isContained(k1, AB) ∧ isContained(k1, CD)
12:  nonQuadCond2 ← isContained(k2, AC) ∧ isContained(k2, BD)
13:  quadNonConvCond1 ← isContained(k1, AB) ∨ isContained(k1, CD)
14:  quadNonConvCond2 ← isContained(k2, AC) ∨ isContained(k2, BD)
15:  if nonQuadCond1 ∨ nonQuadCond2 then
16:    if hasCurveEnvelope(P, S, k1, k2, nonQuadCond1) then
17:      GameType ← ENV_NON_QUAD
18:    else
19:      GameType ← LINEAR_NON_QUAD
20:    end if
21:  else
22:    if quadNonConvCond1 ∧ quadNonConvCond2 then
23:      GameType ← NON_CONVEX_QUAD
24:    else
25:      GameType ← CONVEX_QUAD
26:    end if
27:  end if
28:  return GameType
29: end function

30: function HASCURVEENVELOPE(P, S, k1, k2, nonQuadCond1)
31:  if nonQuadCond1 then
32:    abRatio ← 1 - (distance(a, k1)/distance(a, b))
33:    cdRatio ← 1 - (distance(c, k1)/distance(c, d))
34:    if abRatio == cdRatio then
35:      return false
36:    else
37:      return true
38:    end if
39:  else ▷ nonQuadCond2 holds: same check with AC and BD
40:    ...
41:  end if
42: end function

```

---

**Algoritmo 2.2** Calcolo di  $\partial Imm(\mathbf{u})$ 


---

```

1: function COMPUTEBORDER(M)
2:   type  $\leftarrow$  get2x2GameType(M)
3:   if type == CONVEX_QUAD
4:      $\vee$  type == LINEAR_NON_QUAD  $\vee$  ... then
5:     Border  $\leftarrow$  S
6:   else if type == NON_CONVEX_QUAD then
7:     Border  $\leftarrow$  computeNonConvexQuadBorder(M)
8:   else if type == ENV_NON_QUAD then
9:     Border  $\leftarrow$  computeNonQuadBorder(M)
10:  else
11:    ... ▷ non quadrilateral other degenerate cases
12:  end if
13:  return Border
14: end function

14: function COMPUTENONCONVEXQUADBORDER(M)
▷ quadNonConvCond1 AND quadNonConvCond1 must hold
15:  envEq  $\leftarrow$  0 == diff(u1, x) · diff(u2, y) - diff(u1, y) · diff(u2, x)
16:  invX(y)  $\leftarrow$  solve(envEq, x)
17:  if isContained(ab, k1) then
18:    abRatio  $\leftarrow$  1 - (distance(a, k1)/distance(a, b))
19:    invK1  $\leftarrow$  (abRatio) · c + (1 - abRatio) · d
20:    if isContained(ac, k2) then
21:      acRatio  $\leftarrow$  1 - (distance(a, k2)/distance(a, c))
22:      invK2  $\leftarrow$  (acRatio) · b + (1 - acRatio) · d
23:      Border  $\leftarrow$  {AB, AC, CinvK1, BinvK2}
24:      Border  $\leftarrow$  Border  $\cup$  envPiece(u1(invX(y), y), u2(invX(y), y), 0, abRatio)
25:    else
26:      ...
27:    end if
28:  else
29:    ...
30:  end if
31: end function

32: function COMPUTENONQUADBORDER(M)
▷ nonQuadCond1 OR nonQuadCond2 must hold
33:  envEq  $\leftarrow$  0 == diff(u1, x) · diff(u2, y) - diff(u1, y) · diff(u2, x)
34:  invX(y)  $\leftarrow$  solve(envEq, x)
35:  if nonQuadCond1 then
36:    invK11  $\leftarrow$  (cdRatio) · a + (cdRatio) · b
37:    invK12  $\leftarrow$  (abRatio) · c + (1 - abRatio) · d
38:    if abRatio > cdRatio then
39:      Border  $\leftarrow$  {AC, BD, Ak1, BinvK11, CinvK12, Dk1}
40:      Border  $\leftarrow$ 
      Border  $\cup$  envPiece(u1(invX(y), y), u2(invX(y), y), cdRatio, abRatio)
41:    else
42:      ...
43:    end if
44:  else ▷ nonQuadCond2 holds
45:    ...
46:  end if
47: end function

```

---

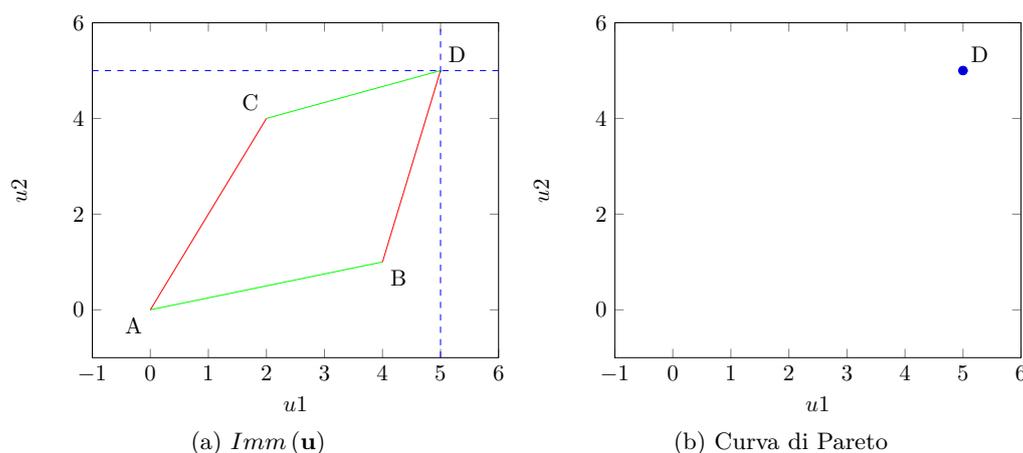


Figura 2.8: PC 1 vettore di payoffs puri.

## 2.2 La Curva di Pareto

Nella Sezione 2.1 è stata fornita una classificazione dei giochi bimatrice  $2 \times 2$ , e si è mostrato che calcolare  $\partial Imm(\mathbf{u})$  è facile. In questa sezione ci si focalizzerà sulla Curva di Pareto di tali giochi: sfruttando i risultati riguardanti  $\partial Imm(\mathbf{u})$  verrà fornita una classificazione esaustiva di tutte le possibili curve di Pareto di giochi  $2 \times 2$ , rispetto alle proprietà geometriche di essa. In particolare verranno fornite anche le condizioni sulla base delle quali è possibile discriminare le varie casistiche, permettendo così la definizione di un algoritmo per calcolare la Curva di Pareto di una qualsiasi bimatrice  $2 \times 2$ .

Per poter fare una classificazione, è necessario innanzitutto definire un criterio con il quale distinguere le principali tipologie di tale curva. Un approccio possibile, e probabilmente il più naturale, è quello di considerare il numero di vettori di payoffs puri (i.e. elementi di  $P$ ) *distinti* che rispettano la proprietà di Pareto efficienza, come discriminante. Analizzando la Tabella 2.1 e usando un po' di intuizione geometrica, risulta evidente infatti che avendo predeterminato  $\partial Imm(\mathbf{u})$ , la Curva di Pareto è individuabile tracciando delle rette orizzontali e verticali in corrispondenza di ogni elemento di  $P$  Pareto efficiente, e valutando le intersezioni con  $\partial Imm(\mathbf{u})$ . Ricordiamo a questo proposito che la verifica di Pareto efficienza di un certo vettore di payoffs puri è verificabile in tempo polinomiale rispetto al numero di mosse dei giocatori [15], nel nostro caso quindi possiamo verificare per ogni vettore di payoffs puri se esso sia Pareto efficiente in tempo costante. Questo procedimento sarà facilmente comprensibile attraverso il supporto di esempi grafici.

Iniziamo quindi considerando il caso di 1 vettore payoff puro (distinto), Pareto efficiente. Considerata la geometria di  $\partial Imm(\mathbf{u})$  infatti, non possono esistere casi in cui non ci sia (almeno) un elemento di  $P$  Pareto efficiente.

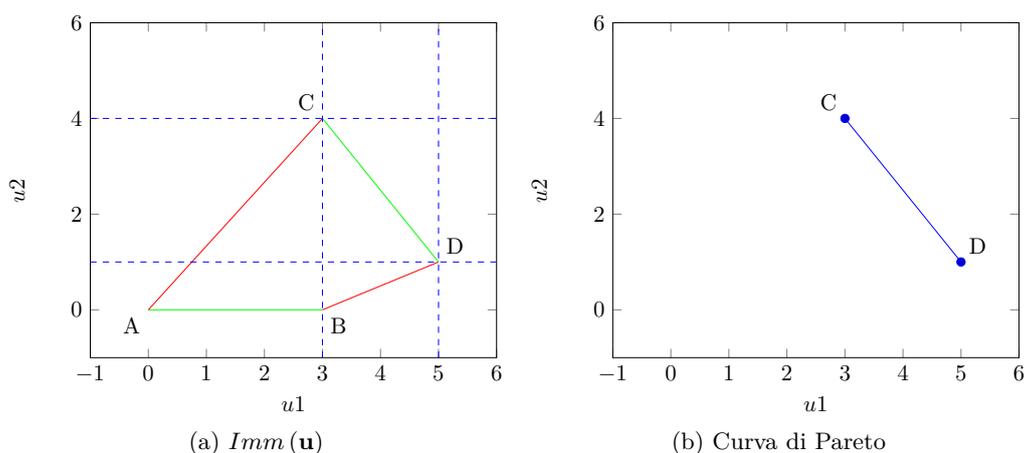


Figura 2.9: PC 2 vettori di payoffs puri: segmento.

### 2.2.1 1 vettore di payoffs puri Pareto efficiente

Se un solo elemento distinto di  $P$  risulta essere Pareto efficiente, questo sarà l'unico punto appartenente alla Curva di Pareto. Questa situazione si può verificare con qualsiasi tipologia di  $Imm(\mathbf{u})$  si consideri. Riportiamo in Figura 2.8  $Imm(\mathbf{u})$  e la Curva di Pareto della bimatrice (2.16), a titolo di esempio. Verificare se esiste soltanto un payoff puro Pareto Efficiente è immediato: esso ha un'ascissa e un'ordinata maggiore (in generale non strettamente) di tutti gli altri payoffs puri. Bisogna ricordare poi che stiamo effettuando una classificazione sulla base dei payoffs puri Pareto efficienti *distinti*. Questo significa un caso in cui due o più payoffs puri collassino nello stesso punto, e siano gli unici ad essere Pareto efficienti, verrà classificato in questa categoria. Non è difficile immaginare situazioni del genere, ad esempio considerando un quadrilatero convesso, degenerato in un triangolo o in un segmento.

### 2.2.2 2 vettori di payoffs puri Pareto efficienti

A differenza del caso descritto in Sezione 2.2.1 in cui la curva era univocamente determinata, nel caso ci siano 2 o più elementi distinti di  $P$  che sono Pareto efficienti,  $P(U)$  può assumere numerose forme, pur avendo sempre 2 vettori di payoffs puri Pareto efficienti. Iniziamo quindi a considerare i casi più semplici, cioè quelli in cui  $P(U)$  non presenta discontinuità. Per la trattazione ritornerà utile definire ora i due vettori di payoffs puri Pareto efficienti distinti: siano  $p = (u_{1p}, u_{2p})$ ,  $s = (u_{1s}, u_{2s})$  gli elementi di  $P$  Pareto efficienti, dove il primo giocatore guadagna di più in  $s$ , mentre il secondo guadagna di più in  $p$  (in altre parole  $u_{2p} > u_{2s}$  e  $u_{1s} > u_{1p}$ ).

### Segmento

Condizione necessaria e sufficiente per questo caso è che i due elementi distinti di  $P$  Pareto efficienti, siano posizionati sulla stessa riga o sulla stessa colonna nella bimatrice considerata. Non ci sono condizioni sulla tipologia di  $Imm(\mathbf{u})$ . Un esempio di questo caso è costituito dalla bimatrice (2.25), rappresentanto nella Figura 2.9.

$$\begin{pmatrix} (0, 0) & (3, 0) \\ (3, 4) & (5, 1) \end{pmatrix} \quad (2.25)$$

### Parabola

La condizione necessaria, ma non sufficiente per questo caso, è che  $Imm(\mathbf{u})$  sia di tipo non quadrilatero degenero, con  $A = D$  o  $B = C$ , e che nessuno dei due vettori di payoffs puri Pareto efficienti sia un elemento della diagonale i cui elementi coincidono. Per ottenere una condizione necessaria e sufficiente, bisogna aggiungere a quella appena esposta, la seguente: sia  $q \in P$  l'elemento distinto *non* Pareto efficiente, cioè il vertice opposto alla curva (ricordiamo che  $q$  rappresenta i due vettori di payoffs identici, su una delle diagonali della bimatrice). Allora deve valere:

$$\begin{aligned} u_{1q} &\geq u_{1p} \\ u_{2q} &\geq u_{2s} \end{aligned} \quad (2.26)$$

Questa condizione descrive la mancanza di discontinuità in  $P(U)$ .

Riassumendo, questo caso si verifica *se e solo se* stiamo considerando il caso in cui 2 elementi di una delle diagonali della bimatrice siano identici, se nessuno dei 2 vettori di payoffs puri sia uno di questi, e se vale la condizione 2.26. La bimatrice soddisfa (2.27) queste condizioni. Il grafico è illustrato in Figura 2.10

$$\begin{pmatrix} (0, 0) & (-0.5, 5) \\ (5, -0.5) & (0, 0) \end{pmatrix} \quad (2.27)$$

### 2 Segmenti

Intuibilmente,  $P(U)$  con 2 vettori di payoffs puri e 2 segmenti, può verificarsi soltanto se  $Imm(\mathbf{u})$  è non quadrilaterale. Infatti i 2 segmenti congiungono i 2 vettori di payoffs puri, al punto di intersezione  $K$  del sistema (2.18), anch'esso elemento della Curva di Pareto. Ciò è necessario ma non sufficiente in generale: per avere una condizione necessaria e sufficiente bisogna che altre 2 proprietà vengano rispettate. La prima, che garantisce l'assenza di discontinuità, è molto simile a quella espressa

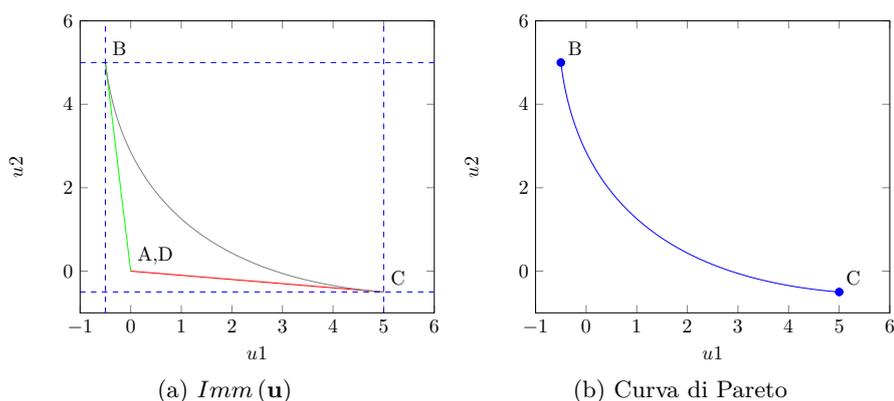


Figura 2.10: PC 2 vettori di payoffs puri: Parabola.

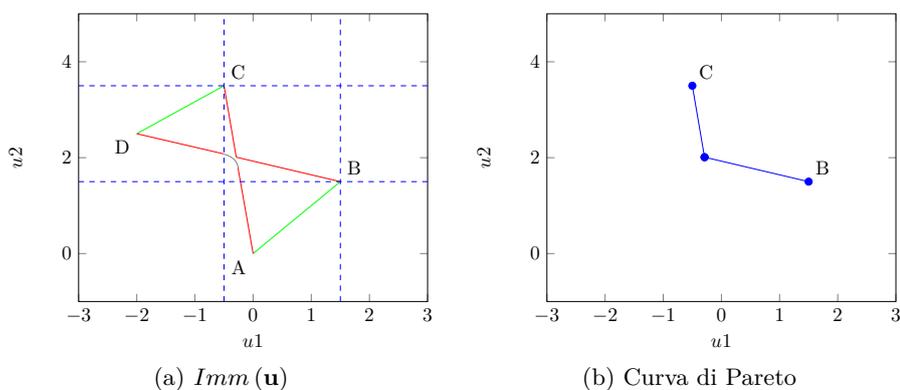


Figura 2.11: PC 2 vettori di payoffs puri: 2 segmenti.

dalle disequazioni (2.26), applicate però questa volta al punto  $K = (u_{1k}, u_{2k})$ :

$$\begin{aligned} u_{1k} &> u_{1p} \\ u_{2k} &> u_{2s} \end{aligned} \tag{2.28}$$

Notiamo che questa volta le disuguaglianze sono strette.

La seconda invece richiede che l'inviluppo (se non è già degenerato in un punto) sia tangente ai segmenti che non toccano i payoff puri Pareto efficienti. In parole povere, cioè, l'inviluppo sia rivolto verso il terzo quadrante.

Non descriviamo questa condizione con formule in quanto sarebbe inutilmente verbosa. Inoltre conoscendo già  $\partial Imm(\mathbf{u})$ , tale condizione è semplice da verificare. La bimatrice (2.29) soddisfa queste condizioni, mentre il grafico è riportato in Figura 2.11

$$\begin{pmatrix} (0, 0) & (1.5, 1.5) \\ (-0.5, 3.5) & (-2, 2.5) \end{pmatrix} \tag{2.29}$$

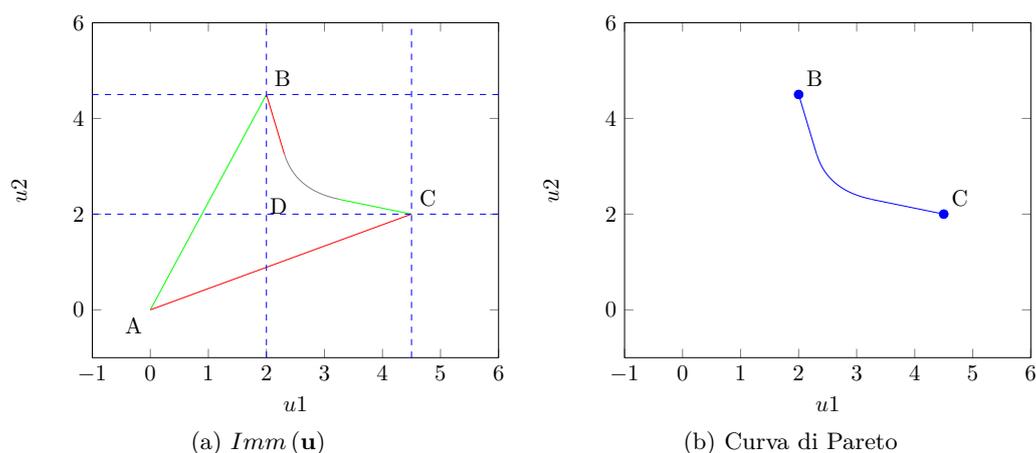


Figura 2.12: PC 2 vettori di payoffs puri: 2 segmenti, parabola.

### Segmento, parabola, segmento

Questa tipologia di Curva di Pareto si può verificare considerando  $Imm(\mathbf{u})$  non quadrilaterale, o quadrilaterale non convesso. Nel primo caso le condizioni necessarie e sufficienti per generare questo caso sono uguali quelle fornite nel caso di 2 vettori di payoffs puri con 2 segmenti, con una differenza: la condizione sull'involuppo è esattamente speculare, cioè l'involuppo deve essere rivolto verso il primo quadrante. Nel secondo invece oltre a richiedere che i 2 vettori di payoffs puri Pareto efficienti non siano sulla stessa riga o colonna, bisogna esprimere la condizione che esprime la continuità di  $P(U)$ , cioè devono valere le condizioni (2.28) applicate però all'elemento di  $P$  associato all'angolo non convesso. Queste costituiscono le condizioni necessarie e sufficienti per questa casistica. La bimatrice (2.30) è un esempio di quadrilatero non convesso, con questo tipo di Curva di Pareto. Il grafico è illustrato nella Figura 2.12.

$$\begin{pmatrix} (0, 0) & (2, 4.5) \\ (4.5, 2) & (2.5, 2.5) \end{pmatrix} \quad (2.30)$$

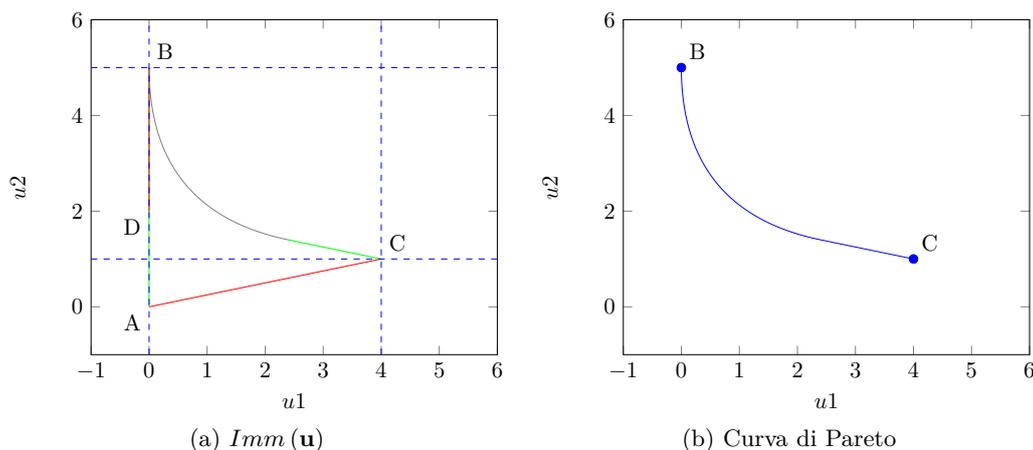


Figura 2.13: PC 2 vettori di payoffs puri: segmento, parabola.

### Segmento, parabola

L'unico caso in cui si può avere una  $P(U)$  di questo tipo si può verificare quando  $Imm(\mathbf{u})$  è di tipo non quadrilatero degenere con 3 vettori di payoffs puri allineati. Dopodichè devono verificarsi due condizioni: non esiste un elemento in  $S$  che congiunga i due payoffs puri Pareto efficienti (altrimenti si avrebbe semplicemente  $P(U)$  costituita da un segmento) e la seguente condizione di continuità:

Sia  $q = K = (u_{1k}, u_{2k})$  il punto di intersezione del sistema (2.18) ed elemento di  $P$ , allineato tra  $p$  e l'ultimo payoff puro rimasto. Allora, affinché la Curva di Pareto considerata sia continua devono valere:

$$\begin{aligned} u_{1k} &\geq u_{1p} \\ u_{2k} &> u_{2s} \end{aligned} \quad (2.31)$$

Notiamo l'asimmetricità delle condizioni: in un caso la disuguaglianza è stretta, nell'altro no. Questo riflette il fatto che in un vertice è la curva ad essere incidente, nell'altro il segmento. Abbiamo implicitamente supposto che  $q$  fosse allineato tra  $p$  e l'ultimo payoff puro rimasto. Se invece fosse allineato tra  $s$  e l'ultimo payoff puro, varrebbero le condizioni speculari a quelle sopra esposte, ottenibili immaginando di scambiare gli assi.

**Esempio** Bimatrice (2.32), Figura 2.13

$$\begin{pmatrix} (0, 0) & (0, 5) \\ (4, 1) & (0, 2) \end{pmatrix} \quad (2.32)$$

**Segmento, 1 discontinuità**

Questa tipologia di  $P(U)$  si può verificare sia nel caso  $Imm(\mathbf{u})$  quadrilatero non convesso, sia in quello non quadrilatero. Com'era logico aspettarsi, la Curva di Pareto non è necessariamente continua, e questo è il primo caso in cui si verifica una discontinuità. Le condizioni che garantivano la continuità nei casi precedenti verranno quindi violate. Proseguendo nella trattazione sarà chiaro come molte forme di  $P(U)$  infatti, possano verificarsi sia nel caso non quadrilatero che in quello quadrilatero non convesso e che le condizioni per entrambe le situazioni siano le stesse: in un caso applicate al punto  $K$  del sistema (2.18), nell'altro applicate al payoff puro di  $P$  associato all'angolo non convesso. Per esprimere le condizioni necessarie e sufficienti affinché  $P(U)$  abbia questa forma considereremo quindi solo il caso non quadrilatero, tenendo bene a mente quanto appena detto. Come condizione necessaria, non deve esistere un elemento di  $S$  che congiunga  $p$  ed  $s$ . Inoltre sia  $K = (u_{1k}, u_{2k})$  il punto di intersezione del sistema (2.18) devono valere le condizioni che descrivono la presenza di una discontinuità:

$$\begin{aligned} u_{1k} &> u_{1p} \\ u_{2k} &< u_{2s} \end{aligned} \tag{2.33}$$

Queste condizioni descrivono la presenza di una discontinuità, rispetto al payoff puro  $s$ : naturalmente le stesse condizioni potrebbero essere espresse in maniera speculare per descriverla rispetto a  $p$ . Una volta garantita la presenza della discontinuità (le condizioni appena espresse sono necessarie e sufficienti per essa) bisogna esprimere un'ultima condizione per assicurarsi che il tratto di curva tangente al segmento incidente a  $p$  sia completamente al di sotto di  $u_{2s}$ . Sia  $p_{inv} = (u_{1p_{inv}}, u_{2p_{inv}})$  il punto estremo del tratto di parabola in cui essa è tangente all'elemento di  $S$  incidente in  $p$ . Questo punto è noto ancora prima di avere calcolato  $\partial Imm(\mathbf{u})$ , grazie al sistema (2.18). Allora, la condizione affinché sia presente solo un segmento prima della discontinuità è:

$$u_{2p_{inv}} \leq u_{2s} \tag{2.34}$$

La congiunzione di tutte queste condizioni è necessaria e sufficiente per questa tipologia di Curva di Pareto<sup>5</sup>.

**Esempio** Bimatrice (2.35), Figura 2.14

$$\begin{pmatrix} (0, 0) & (2, 4) \\ (4, 2) & (3, 1) \end{pmatrix} \tag{2.35}$$

---

<sup>5</sup>Non sono stati citati i casi non quadrilaterali degeneri per evitare di essere verbosi. Illustrato il caso generale, il buon senso è sufficiente a comprendere le eventuali degenerazioni.

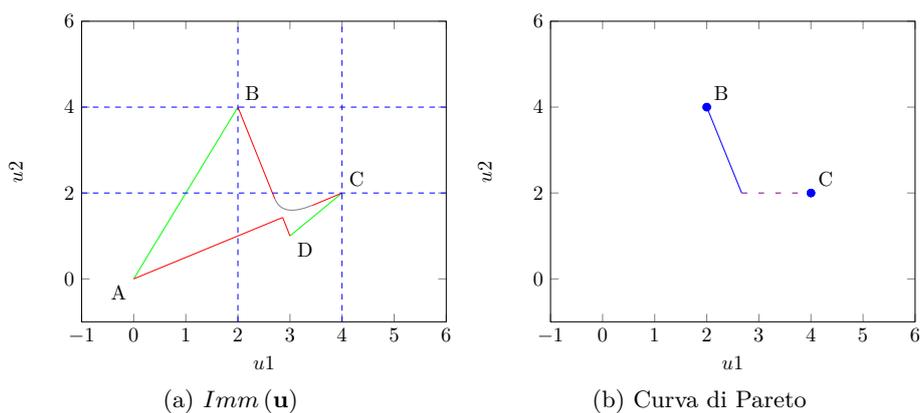


Figura 2.14: PC 2 vettori di payoffs puri: segmento, 1 discontinuità.

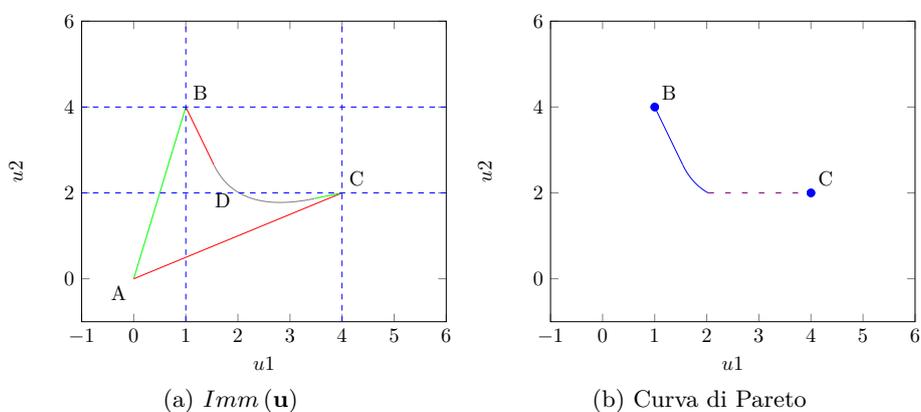


Figura 2.15: PC 2 vettori di payoffs puri: segmento, parabola, 1 discontinuità.

### Segmento, parabola, 1 discontinuità

Il lettore più attento avrà intuito che questo il caso ed il precedente sono in gran parte sovrapponibili. In effetti valgono esattamente le stesse condizioni del caso segmento con discontinuità. Se prima era stato necessario esplicitare il fatto che la curva fosse al di sotto della discontinuità, in questo dobbiamo asserire esattamente l'opposto. Valgono quindi esattamente le stesse condizioni del caso precedente, tranne la (2.34) che viene sostituita dalla seguente:

$$u_{2pinv} > u_{2s}$$

**Esempio** Bimatrice 2.36, Figura 2.15

$$\begin{pmatrix} (0, 0) & (1, 4) \\ (4, 2) & (2, 1.5) \end{pmatrix} \tag{2.36}$$

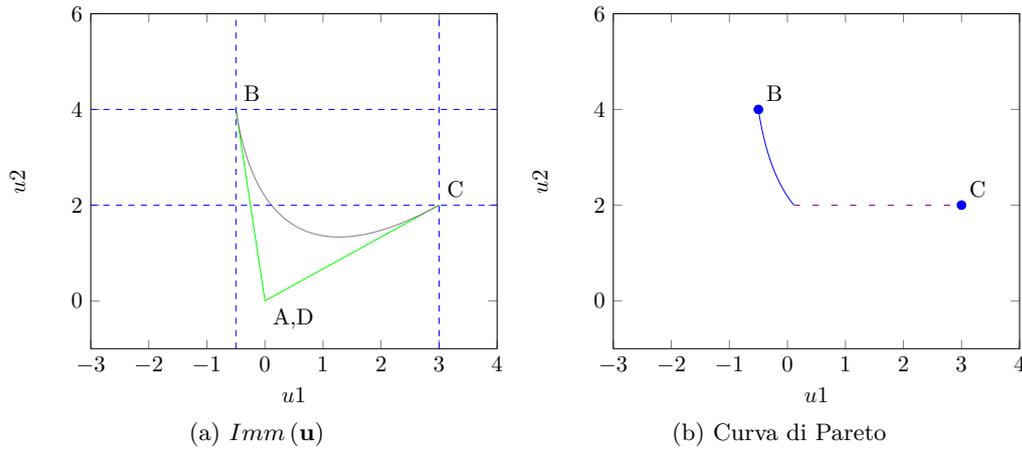


Figura 2.16: PC 2 vettori di payoffs puri: parabola, discontinuità.

### Parabola, 1 discontinuità

Affinche si possa verificare questo caso è necessario che  $Imm(\mathbf{u})$  sia non quadrilaterale degenera (3 payoffs allineati o 2 identici su una diagonale della bimatrice). Nel caso di 2 payoffs identici su una diagonale della bimatrice, oltre alla banale condizione per cui non esiste un elemento di  $S$  che congiunga  $p$  ed  $s$ , la seguente condizione è necessaria e sufficiente per questa casistica: sia  $q \in P$  l'elemento distinto *non* Pareto efficiente, cioè il vertice opposto alla curva:

$$\begin{aligned} u_{1q} &< u_{1p} \\ u_{2q} &\geq u_{2s} \end{aligned} \quad (2.37)$$

Anche in questo caso queste condizioni possono essere formulate in modo speculare per variare il payoff puro Pareto efficiente isolato. Nel caso invece di 3 payoffs allineati le condizioni sono in qualche modo simili. Oltre alla condizione banale di non congiunzione lineare tra  $p$  ed  $s$ , sia  $q = K = (u_{1k}, u_{2k})$  il punto di intersezione del sistema (2.18) allineato tra  $p$  ed  $q \in P$  non Pareto efficiente, deve valere:

$$\begin{aligned} u_{1k} &> u_{1p} \\ u_{2k} &\leq u_{2s} \end{aligned} \quad (2.38)$$

O l'analoga versione speculare.

**Esempio** Bimatrice (2.39), Figura 2.16

$$\begin{pmatrix} (0, 0) & (-0.5, 4) \\ (3, 2) & (0, 0) \end{pmatrix} \quad (2.39)$$

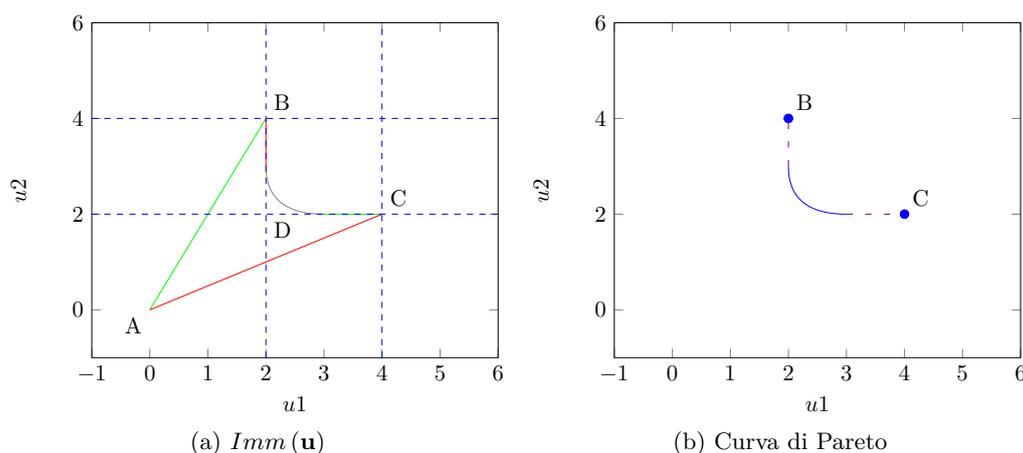


Figura 2.17: PC 2 vettori di payoffs puri: parabola, 2 discontinuità.

### Parabola, 2 discontinuità

Non ci sono particolari vincoli *a priori* sul tipo di  $Imm(\mathbf{u})$  necessaria affinché questo caso possa verificarsi, se non quello che deve esistere un tratto di parabola tra  $p$  ed  $s$ . Si prenderà come riferimento il caso quadrilaterale non convesso esprimendo le due condizioni che prese congiuntamente sono necessarie e sufficienti per questa casistica. Con lievi variazioni e guidati dalla ragionevolezza, gli altri casi sono derivabili con lievi variazioni delle condizioni esposte. La prima condizione esprime la presenza di 2 discontinuità: sia  $q$  il payoff puro associato all'angolo non convesso, allora esistono due discontinuità se e solo se:

$$\begin{aligned} u_{1q} &\leq u_{1p} \\ u_{2q} &\leq u_{2s} \end{aligned} \quad (2.40)$$

La seconda invece esprime la presenza di un tratto di inviluppo totalmente separato da  $p$  e da  $s$  nella Curva di Pareto. Sia  $u_2 = f_{inv}(u_1)$  la funzione che descrive il tratto di parabola. Allora se  $f_{inv}$  è definita in  $u_{1p}$  e se vale:

$$f_{inv}(u_{1p}) \geq u_{2s}$$

Esiste un tratto di curva (eventualmente degenerato in un punto) appartenente a  $P(U)$ .

**Esempio** Bimatrice (2.41), Figura 2.17

$$\begin{pmatrix} (0, 0) & (2, 4) \\ (4, 2) & (2, 2) \end{pmatrix} \quad (2.41)$$



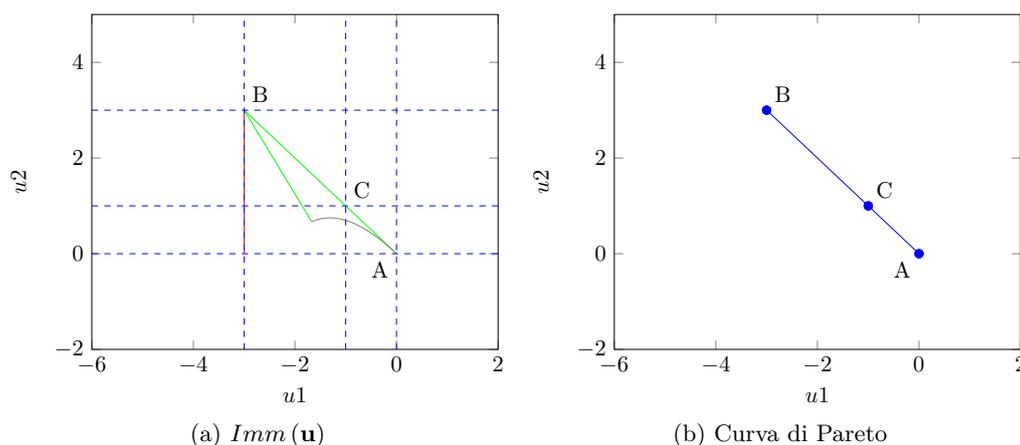


Figura 2.19: PC 3 vettori di payoffs puri: segmento.

### 2.2.3 3 vettori di payoffs puri Pareto efficienti

Analizziamo ora il caso in cui 3 vettori di payoffs puri appartengano alla Curva di Pareto. Come nel caso in cui solo due vettori di payoffs puri risultavano Pareto efficienti, anche in questo abbiamo numerose forme possibili. Definiamo quindi i tre vettori di payoffs puri Pareto efficienti distinti: siano  $p = (u_{1p}, u_{2p})$ ,  $q = (u_{1q}, u_{2q})$ ,  $s = (u_{1s}, u_{2s})$  gli elementi di  $P$  Pareto efficienti, con  $u_{1p} < u_{1q} < u_{1s}$ .

#### Segmento

$P(U)$  con 1 segmento e 3 vettori di payoffs puri distinti può essere ottenuta con  $Imm(\mathbf{u})$  non quadrilaterale degenere (3 vettori di payoffs puri allineati), o con il caso quadrilaterale degenerato in un segmento. Nel primo caso è sufficiente che il segmento congiunga  $p$  ed  $s$ , e  $q$  sia allineato tra i 2. Nel secondo invece affinché il numero di vettori di payoffs puri distinti rimanga tre, è necessario (e sufficiente) che due vettori di payoffs puri qualsiasi coincidano.

**Esempio** Bimatrice (2.43), Figura 2.19

$$\begin{pmatrix} (0, 0) & (-3, 3) \\ (-1, 1) & (-3, 0) \end{pmatrix} \quad (2.43)$$

#### 2 Segmenti

Questo caso è uno dei più immediati: deve esistere un elemento in  $S$  che congiunga  $p$  e  $q$  ed un altro che congiunga  $q$  ed  $s$ . Questo si può verificare soltanto se  $Imm(\mathbf{u})$  è quadrilaterale, o se è non quadrilaterale con 2 vettori di payoffs puri identici sulla stessa diagonale (nel qual caso sarebbero 4 i vettori di payoffs puri Pareto efficienti, di cui solo 3 distinti).

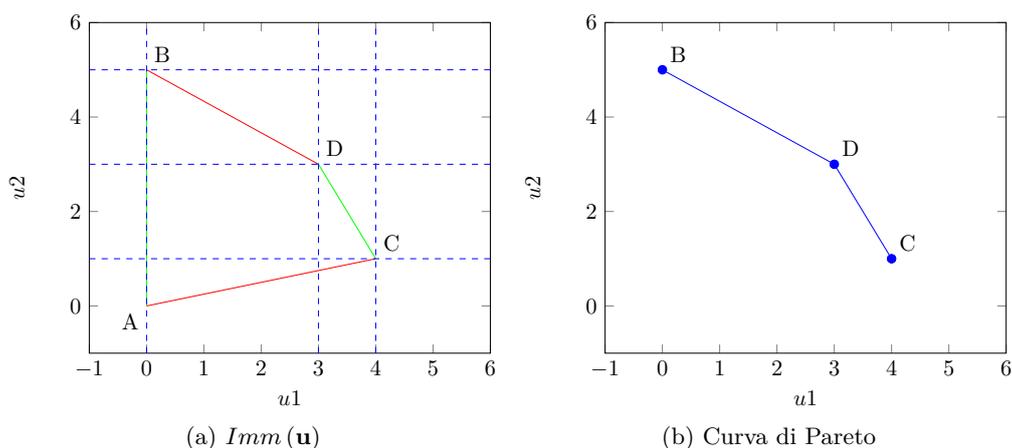


Figura 2.20: PC 3 vettori di payoffs puri: 2 segmenti.

**Esempio** Bimatrice (2.44), Figura 2.20

$$\begin{pmatrix} (0, 0) & (0, 5) \\ (4, 1) & (3, 3) \end{pmatrix} \quad (2.44)$$

**2 Segmenti, parabola** La situazione in cui  $P(U)$  è costituita da due segmenti, un tratto di curva e tre vettori di payoffs puri si può verificare unicamente se  $Imm(\mathbf{u})$  è di tipo non quadrilatero degenere, con 3 vettori di payoffs puri allineati. Anche in questo caso è semplice esprimere la condizione che da origine a questo tipo di Curva di Pareto: deve esistere un (e uno solo) elemento di  $S$  che congiunga due dei tre vettori di payoffs puri Pareto efficienti. Supponiamo quindi che  $p$  e  $q$  siano uniti attraverso un segmento. Sia allora  $K = (u_{1k}, u_{2k})$  il punto di intersezione del sistema (2.18) deve valere:

$$u_{1k} > u_{1q}$$

Che descrive la continuità della Curva di Pareto.

**Esempio** Bimatrice (2.45), Figura 2.21

$$\begin{pmatrix} (0, 0) & (1, -1) \\ (-1, 2.5) & (-3, 3) \end{pmatrix} \quad (2.45)$$

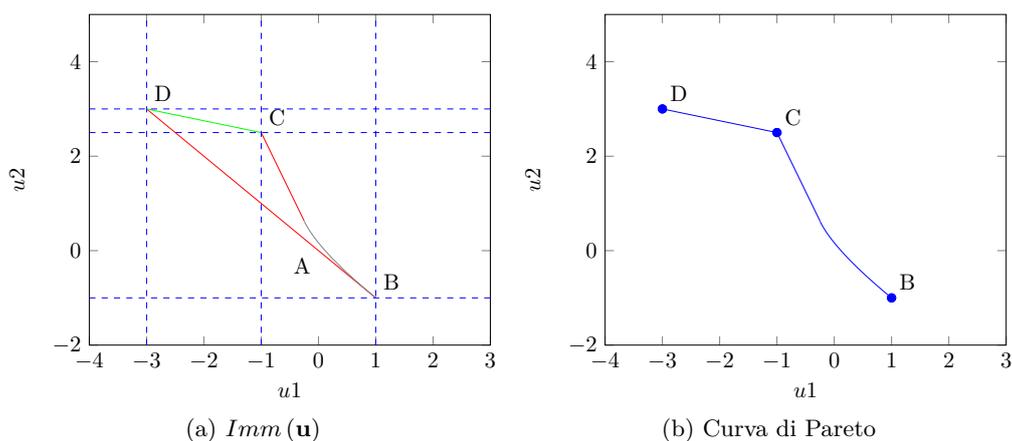


Figura 2.21: PC 3 vettori di payoffs puri: 2 segmenti, parabola.

### 3 Segmenti

Esattamente per l'analogo caso con 2 vettori di payoffs puri Pareto efficienti, è possibile avere lo stesso numero di segmenti rispetto al numero di vettori di payoffs puri nella Curva di Pareto, se e solo se  $Imm(\mathbf{u})$  è non quadrilaterale (eventualmente lineare a tratti, ma non degenerare). Inoltre deve esistere un (e uno solo) elemento di  $S$  che congiunga due dei tre vettori di payoffs puri Pareto efficienti. Suppiano quindi che  $p$  e  $q$  siano uniti attraverso un segmento. Sia allora  $K = (u_{1k}, u_{2k})$  il punto di intersezione del sistema (2.18) deve valere:

$$u_{1k} > u_{1q} \tag{2.46}$$

Che garantisce la continuità della Curva di Pareto, ed inoltre ispezionando  $\partial Imm(\mathbf{u})$  non deve esistere l'involuppo tra  $q$  ed  $s$  (ma eventualmente tra  $p$  e l'ultimo elemento appartenente a  $P$ ).

**Esempio** Bimatrice (2.46), Figura 2.22

$$\begin{pmatrix} (0, 0) & (2, -0.3) \\ (4, -2) & (2.7, -4) \end{pmatrix} \tag{2.47}$$

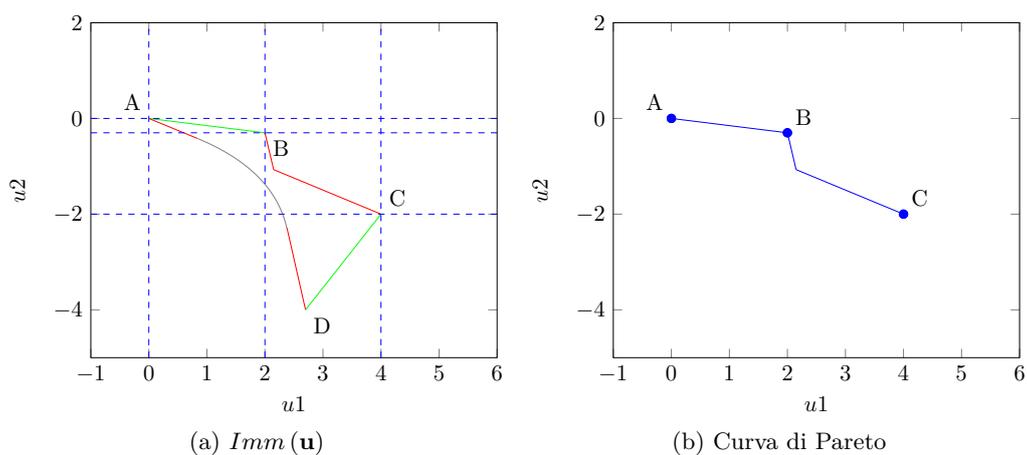


Figura 2.22: PC 3 vettori di payoffs puri: 3 segmenti.

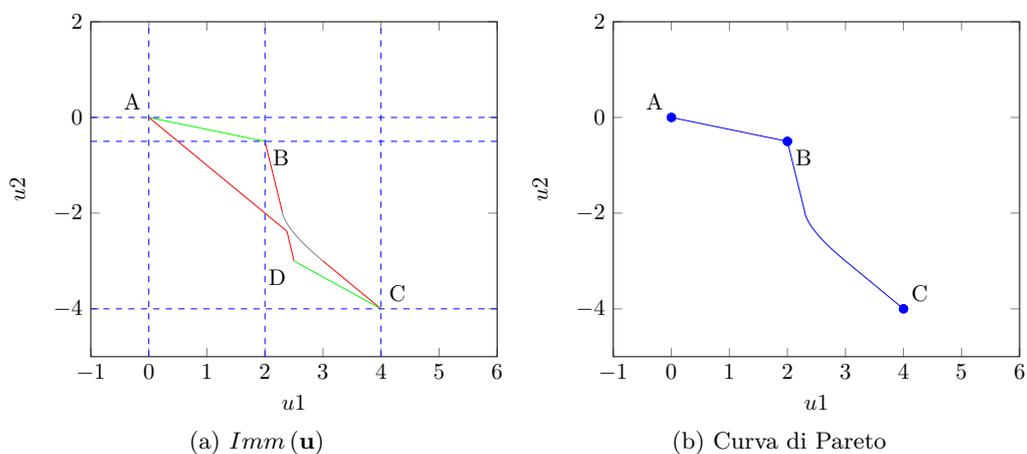


Figura 2.23: PC 3 vettori di payoffs puri: 3 segmenti, parabola.

### 3 Segmenti, parabola

Valgono le stesse identiche condizioni e requisiti del caso precedente, con l'intuibile differenza che deve necessariamente esistere un inviluppo non degenere, e questo debba esistere tra  $q$  ed  $s$ .

**Esempio** Bimatrice (2.48), Figura 2.23

$$\begin{pmatrix} (0, 0) & (2, -0.5) \\ (4, -4) & (2.5, -3) \end{pmatrix} \tag{2.48}$$

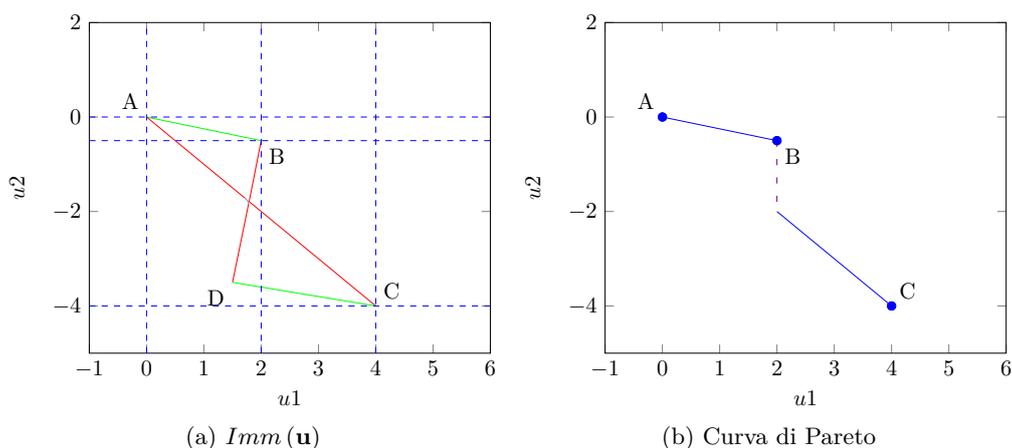


Figura 2.24: P.C. 3 payoffs puri: 2 segmenti, 1 discontinuità

## 2 Segmenti, 1 discontinuità

Questo caso è derivato da quello con 3 segmenti, rimuovendo la condizione di continuità. Si sta considerando quindi  $Imm(\mathbf{u})$  non quadrilaterale (eventualmente lineare a tratti, ma non degenerare) in cui i vettori di payoffs puri Pareto efficienti  $p$  e  $q$  sono uniti attraverso un segmento e non esista un tratto d'involuppo tra  $q$  ed  $s$ , né un segmento che li congiunga direttamente. La condizione (2.46) invece viene sostituita con la sua complementare:

$$u_{1k} \leq u_{1q}$$

Che esprime la presenza di una discontinuità. Inoltre è possibile allentare la condizione di non esistenza di un tratto di involuppo tra  $q$  ed  $s$ , a patto che si verifichi la seguente condizione: sia  $p_{inv} = (u_{1p_{inv}}, u_{2p_{inv}})$  il punto estremo del tratto di parabola in cui essa è tangente all'elemento di  $S$  incidente in  $s$ . Allora deve valere:

$$u_{1p_{inv}} \leq u_{1q}$$

**Esempio** Bimatrice (2.49), Figura 2.24

$$\begin{pmatrix} (0, 0) & (2, -0.5) \\ (4, -4) & (1.5, -3.5) \end{pmatrix} \quad (2.49)$$

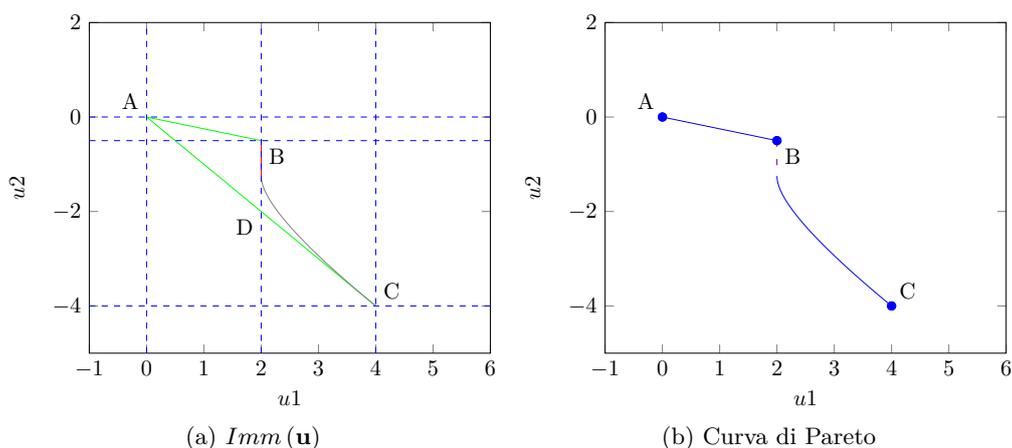


Figura 2.25: PC 3 vettori di payoffs puri: segmento, parabola, 1 discontinuità.

### Segmento, parabola, 1 discontinuità

Come probabilmente sarà già stato palesato, il caso non quadrilaterale è quello che da origine al maggior numero di Curve di Pareto possibili. Affinchè esista un segmento ed un tratto di curva con una discontinuità, è necessario che  $Imm(\mathbf{u})$  sia non quadrilaterale degenere con 3 vettori di payoffs puri allineati. Inoltre i vettori di payoffs puri  $p$  e  $q$  devono essere congiunti da un segmento. A questo punto supponiamo che la discontinuità si verifichi in corrispondenza dell'ascissa  $u_{1q}$ , cioè:

$$u_{1k} \leq u_{1q}$$

Affinchè il tratto di parabola tocchi  $s$  e non  $q$  bisogna esprimere l'ultima condizione: il punto  $r = K$  del sistema (2.18) dev'essere allineato tra  $p$  ed  $s$  e non tra  $p$  e  $q$ .

**Esempio** Bimatrice (2.50), Figura 2.25

$$\begin{pmatrix} (0, 0) & (2, -0.5) \\ (4, -4) & (2, -2) \end{pmatrix} \quad (2.50)$$

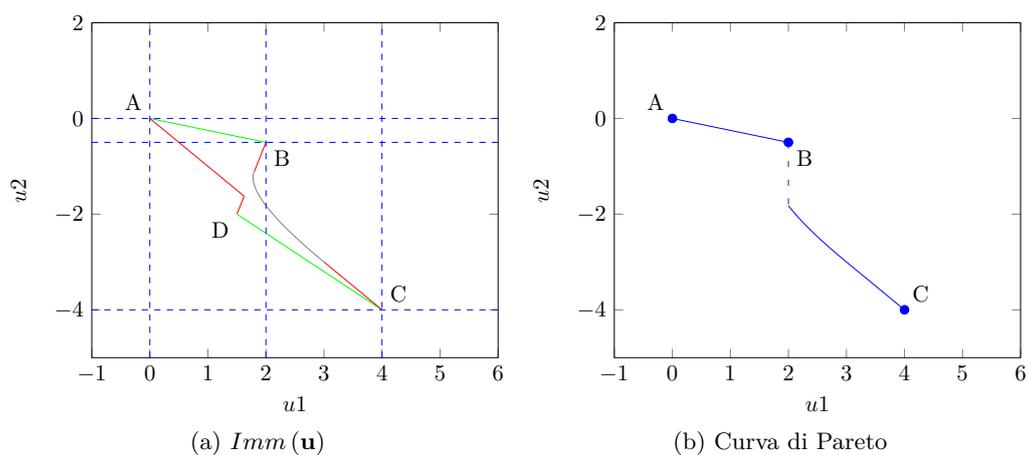


Figura 2.26: PC 3 vettori di payoffs puri: 2 segmenti, parabola, 1 discontinuità.

## 2 Segmenti, parabola, 1 discontinuità

Questa tipologia è esattamente identica a quella analoga senza la presenza del tratto di parabola. L'unica differenza è legata ovviamente alla condizione sulla presenza dell'involuppo: in questo caso deve esistere l'involuppo tra  $q$  ed  $s$ . e deve anche rispettare la seguente condizione: Sia  $p_{inv} = (u_{1p_{inv}}, u_{2p_{inv}})$  il punto estremo del tratto di parabola in cui essa è tangente all'elemento di  $S$  incidente in  $s$ . Allora deve valere:

$$u_{1p_{inv}} > u_{1q}$$

**Esempio** Bimatrice (2.51), Figura 2.26

$$\begin{pmatrix} (0, 0) & (2, -0.5) \\ (4, -4) & (1.5, -2) \end{pmatrix} \quad (2.51)$$

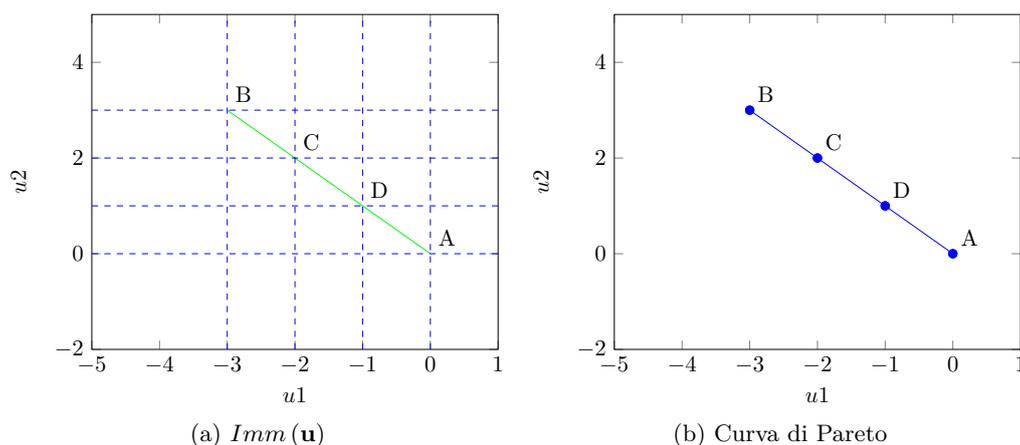


Figura 2.27: PC 4 vettori di payoffs puri: segmento.

#### 2.2.4 4 vettori di payoffs puri Pareto efficienti

Questa tipologia di Curva di Pareto è possibile soltanto nei casi in cui  $Imm(\mathbf{u})$  sia non quadrilaterale, e rappresenta dei casi piuttosto «distorti» dal punto di vista geometrico. Definiamo quindi i quattro vettori di payoffs puri Pareto efficienti distinti: siano  $p = (u_{1p}, u_{2p})$ ,  $q = (u_{1q}, u_{2q})$ ,  $r = (u_{1r}, u_{2r})$ ,  $s = (u_{1s}, u_{2s})$  gli elementi di  $P$  Pareto efficienti, con  $u_{1p} < u_{1q} < u_{1r} < u_{1s}$ .

##### Segmento

Si potrebbe definire questo caso come degenero. È realizzabile se e solo se i quattro vettori di payoffs puri sono tutti distinti, allineati tra loro e Pareto efficienti. In questo caso  $Imm(\mathbf{u}) = \partial Imm(\mathbf{u}) = P.C.$

**Esempio** Bimatrice (2.52), Figura 2.30

$$\begin{pmatrix} (0, 0) & (-3, 3) \\ (-2, 2) & (-1, 1) \end{pmatrix} \quad (2.52)$$

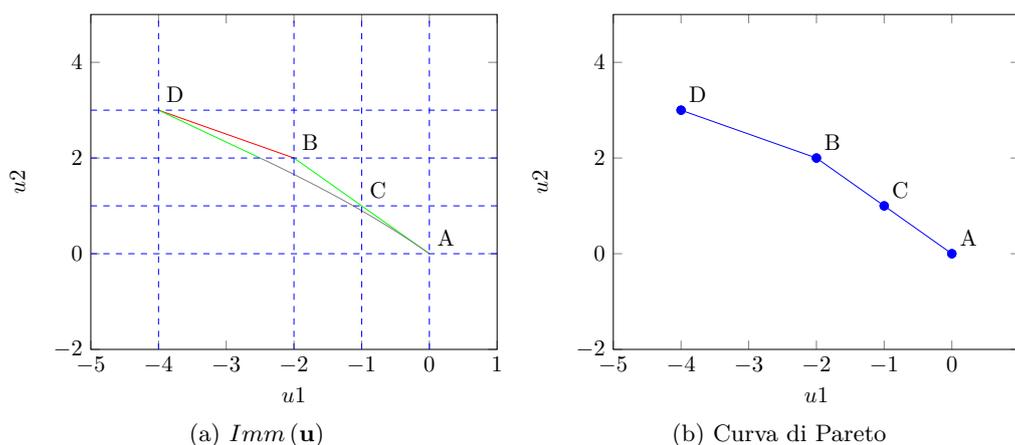


Figura 2.28: P.C. 4 payoffs puri: 2 segmenti

## 2 Segmenti

Condizione necessaria e sufficiente affinché si verifichi questo caso è che esista un segmento di  $S$  che congiunga  $p$  e  $q$ , uno che congiunga  $q$  ed  $s$ , e che  $r$  sia allineato tra  $q$  ed  $s$ . Come in tutti i casi precedenti, queste condizioni possono essere formulate nell'analogia versione speculare. Ci si accorge immediatamente che questa casistica si può verificare solo con  $Imm(\mathbf{u})$  non quadrilaterale degenera con 3 vettori di payoffs puri allineati.

**Esempio** Bimatrice (2.53), Figura 2.30

$$\begin{pmatrix} (0,0) & (-2,2) \\ (-1,1) & (-4,3) \end{pmatrix} \quad (2.53)$$

## 3 Segmenti

Condizione necessaria e sufficiente affinché si verifichi questo caso è che  $Imm(\mathbf{u})$  sia quadrilaterale convessa, con 4 vettori di payoffs puri Pareto efficienti.

**Esempio** Bimatrice (2.54), Figura 2.29

$$\begin{pmatrix} (0,0) & (1,-0.5) \\ (2.5,-2.5) & (2,-1.5) \end{pmatrix} \quad (2.54)$$

## 4 Segmenti

Abbiamo già accennato al fatto che  $P(U)$  con 4 vettori di payoffs puri sono possibili soltanto quando  $Imm(\mathbf{u})$  è non quadrilaterale. Lo specifico caso con quattro

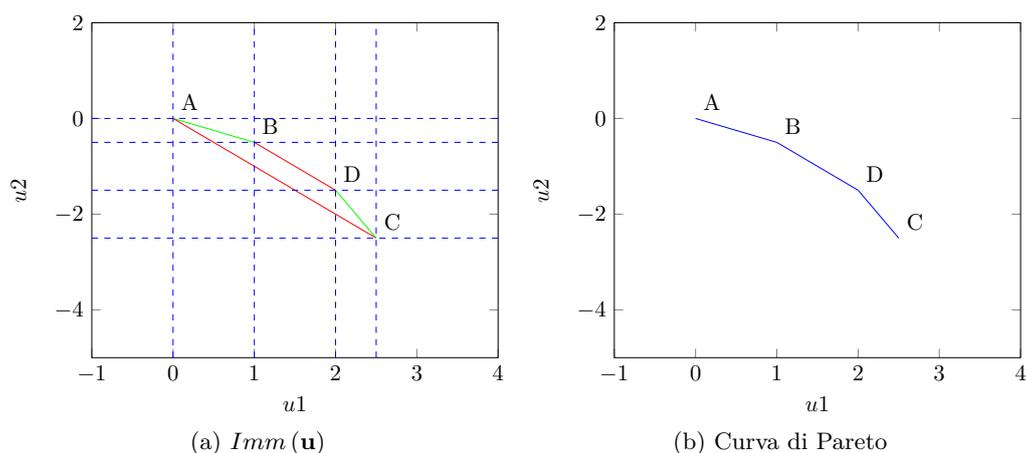


Figura 2.29: PC 4 vettori di payoffs puri: 3 segmenti.

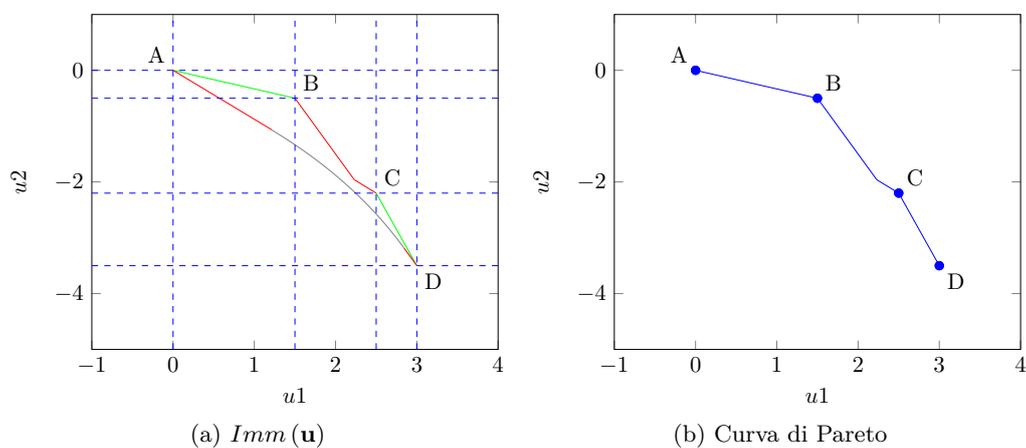


Figura 2.30: PC 4 vettori di payoffs puri: 4 segmenti.

segmenti richiede il soddisfacimento delle seguenti condizioni:  $Imm(\mathbf{u})$  non dev'essere un non quadrilatero degenere (mentre è ammesso quello lineare a tratti), devono esistere due elementi di  $S$  che congiungano  $p, q$  ed  $r, s$  e l'involuppo deve esistere tra i vettori di payoffs puri  $p$  ed  $s$ .

**Esempio** Bimatrice (2.55), Figura 2.30

$$\begin{pmatrix} (0, 0) & (1.5, -0.5) \\ (2.5, -2.2) & (3, -3.5) \end{pmatrix} \tag{2.55}$$

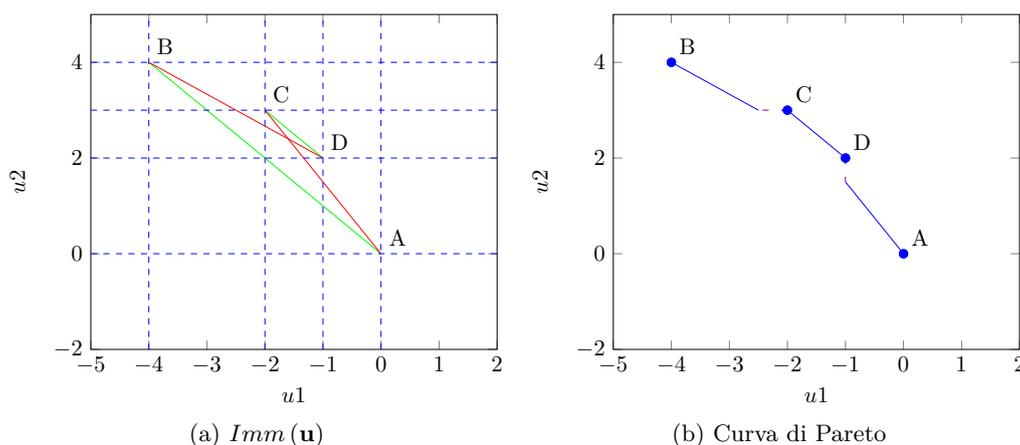


Figura 2.31: PC 4 vettori di payoffs puri: 3 segmenti, 2 discontinuità.

### 3 Segmenti, 2 discontinuità

La tipologia di  $Imm(\mathbf{u})$  più immediata per generare questo caso è quella non quadrilaterale lineare a tratti. Ponendoci in questo caso è sufficiente aggiungere la condizione che esista un elemento di  $S$  che congiunge  $q$  ed  $r$  per ottenere le condizioni necessarie e sufficienti per questa casistica. Oltre a questo tipo di  $Imm(\mathbf{u})$  è possibile generare la Curva di Pareto che stiamo considerando, anche senza richiedere la linearità a tratti. Cioè è possibile che esista un inviluppo tra  $p$  e  $q$  o tra  $r$  ed  $s$ . Supponendo che esista tra  $p$  e  $q$  oltre alle condizioni espresse per il caso lineare a tratti bisogna aggiungere quella che descrive l'assenza dell'inviluppo in  $P(U)$ . Sia  $p_{inv} = (u_{1p_{inv}}, u_{2p_{inv}})$  il punto estremo del tratto di parabola in cui essa è tangente all'elemento di  $S$  incidente in  $p$ . Allora deve valere la la condizione:

$$u_{2p_{inv}} \leq u_{2q} \quad (2.56)$$

**Esempio** Bimatrice (2.57), Figura 2.31

$$\begin{pmatrix} (0,0) & (-4,4) \\ (-2,3) & (-1,2) \end{pmatrix} \quad (2.57)$$

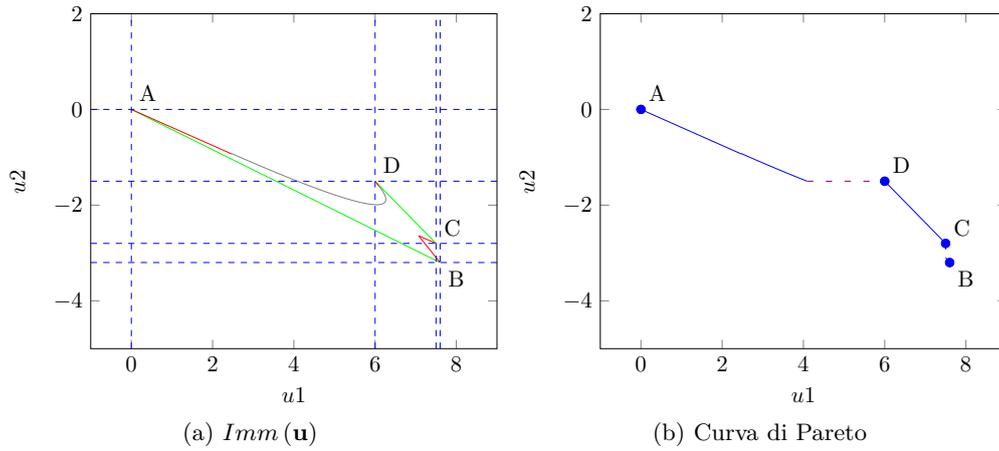


Figura 2.32: PC 4 vettori di payoffs puri: 3 segmenti, parabola, 2 discontinuità.

### 3 Segmenti, parabola, 2 discontinuità

Intuibilmente valgono le stesse condizioni espresse nel caso precedente (3 segmenti con 2 discontinuità) in cui la condizione 2.56 che determina l'assenza dell'involuppo in  $P(U)$  (quando presente) viene complementata:

$$u_{2pinv} > u_{2q}$$

Non vi sono altre possibilità per ottenere questa forma della Curva di Pareto.

**Esempio** Bimatrice (2.58), Figura 2.32

$$\begin{pmatrix} (0, 0) & (7.6, -3.2) \\ (7.5, -2.8) & (6, -1.5) \end{pmatrix} \quad (2.58)$$

### 2 Segmenti, parabola, 1 discontinuità

L'unica tipologia di  $Imm(\mathbf{u})$  che può generare una Curva di Pareto di questo tipo è quella non quadrilaterale degenere con 3 vettori di payoffs puri allineati. Essendo tutti i payoffs puri Pareto efficienti, necessariamente esisterà un segmento appartenente ad  $S$  tra  $p$  e  $r$  (con al proprio interno  $q$ ) o tra  $q$  ed  $s$  (con al proprio interno  $r$ ). Supponiamo quindi che esista un segmento tra  $p$  ed  $r$  e che  $q$  sia allineato tra di loro. Allora affinché un tratto di involuppo faccia parte di  $P(U)$ , sia  $s_{inv} = (u_{1sinv}, u_{2sinv})$  il punto estremo del tratto di parabola in cui essa è tangente all'elemento di  $S$  incidente in  $s$ , deve valere:

$$u_{1sinv} > u_{1r} \quad (2.59)$$

**Esempio** Bimatrice (2.60), Figura 2.33

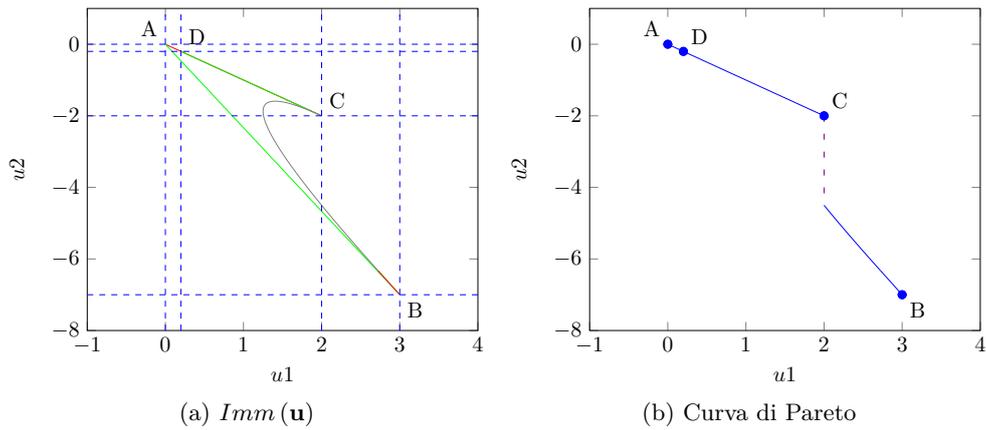


Figura 2.33: P.C. 4 payoffs puri: 2 segmenti, parabola, 1 discontinuità

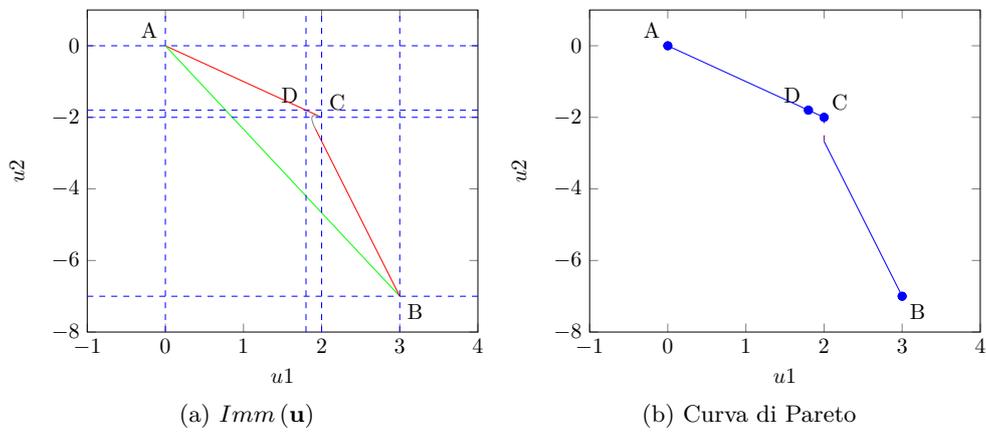


Figura 2.34: PC 4 vettori di payoff puri: 2 segmenti, 1 discontinuità.

$$\begin{pmatrix} (0,0) & (3,-7) \\ (2,-2) & (0.2,-0.2) \end{pmatrix} \quad (2.60)$$

### 2 Segmenti, 1 discontinuità

Valgono le stesse condizioni del caso precedente tranne la (2.59) che viene complementata, diventando:

$$u_{1sinv} \leq u_{1r}$$

**Esempio** Bimatrice (2.61), Figura 2.34

$$\begin{pmatrix} (0,0) & (3,-7) \\ (2,-2) & (1.8,-1.8) \end{pmatrix} \quad (2.61)$$

### 2.2.5 Considerazioni

$P(U)$  di una generica bimatrice  $2 \times 2$   $U$  può assumere un numero di forme non indifferente, ma limitato. Dalla Tabella 2.2 notiamo che può assumere 27 forme diverse. È difficile immaginare una dimostrazione analitica per la completezza di questa analisi, ma gli argomenti proposti dovrebbero essere sufficientemente solidi per rendersi conto che perturbando uno qualsiasi dei casi presentati (o una qualsiasi condizione iniziale scelta) si ricade in una delle 27 forme discusse.

Per ogni caso è stata esposta la relativa condizione necessaria e sufficiente che ne da origine. Avendo inoltre presentato anche l'algoritmo per il calcolo di  $\partial Imm(\mathbf{u})$ , tutte le condizioni sono facilmente verificabili in maniera sistematica. In altre parole strutturando in maniera sequenziale le condizioni esposte, e ricordando che è possibile verificare se un dato vettore di payoffs è Pareto efficiente in tempo polinomiale rispetto al numero di azioni [15] è possibile scrivere un algoritmo per il calcolo *esatto* della Curva di Pareto di un qualsiasi gioco  $2 \times 2$  che determina inizialmente il numero di payoffs puri Pareto efficienti, per poi procedere con la verifica delle possibili conseguenti casistiche.

L'esattezza del calcolo inoltre non dipende in alcun modo dall'insieme numerico a cui appartengono gli elementi della bimatrice originale (irrazionali o razionali). Non potendo rappresentare un numero irrazionale in modo esatto con una precisione numerica finita, parlare di *soluzioni esatte* ha senso solo sottintendendo una rappresentazione di tipo simbolico. Poichè è possibile effettuare il calcolo di  $\partial Imm(\mathbf{u})$  (simbolicamente nel caso di numeri irrazionali) in modo esatto, e la selezione dei tratti Pareto efficienti è effettuata senza effettuare altre operazioni se non quelle di restringimento del dominio di definizione dei singoli tratti, vale il seguente teorema:

**Teorema** Il calcolo *esatto* della Curva di Pareto di un qualsiasi gioco  $2 \times 2$ , ha una complessità computazionale temporale costante.

Tabella 2.2: Numero di PC di una bimatrice  $2 \times 2$

Numero di payoffs puri distinti	Numero di forme possibili
1	1
3	10
3	8
4	8
Qualsiasi	27

Tabella 2.3: Riepilogo Curve di Pareto  $2 \times 2$  (1)

Vettori di payoff puri distinti	Tratti	$Imm(\mathbf{u})$					
		Quad. convesso	Quad non convesso	Non quad.	Non quad. lineare	Non quad. degenerare 1	Non quad. degenerare 2
1	Punto	Degenerare	No	No	No	No	No
2	Segmento	Sì	Sì	Sì	Sì	Sì	Sì
2	Parabola	No	No	No	No	No	Sì
2	Segmento, parabola	No	No	No	No	Sì	No
2	2 segmenti	No	No	Sì	Sì	No	No
2	2 segmenti, parabola	No	Sì	Sì	No	No	No
2	Segmento, 1 discontinuità	No	Sì	Sì	Sì	Sì	No
2	Parabola, 1 discontinuità	No	No	No	No	No	Sì
2	Segmento, parabola, 1 discontinuità	No	Sì	Sì	No	Sì	No
2	Parabola, 2 discontinuità	No	Sì	Sì	Sì	Sì	Sì
2	Punti isolati	No	Sì	Sì	Sì	Sì	Sì

Tabella 2.4: Riepilogo Curve di Pareto  $2 \times 2$  (2)

Vettori di payoffs puri distinti	Tratti	$Imm(\mathbf{u})$					
		Quad. convesso	Quad. non convesso	Non quad.	Non quad. lineare	Non quad. degenerare 1	Non quad. degenerare 2
3	Segmento	Degenerare	No	No	No	No	No
3	2 segmenti	Sì	Sì	No	No	No	Sì
3	2 segmenti, parabola	No	No	No	No	Sì	No
3	3 segmenti	No	No	Sì	Sì	No	No
3	3 segmenti, parabola	No	No	Sì	No	No	No
3	Segmento, parabola, 1 discontinuità	No	No	No	No	Sì	No
3	2 segmenti, 1 discontinuità	No	No	Sì	Sì	No	No
3	2 segmenti, parabola, 1 discontinuità	No	No	No	No	No	Sì

Tabella 2.5: Riepilogo Curve di Pareto  $2 \times 2$  (3)

Vettori di payoffs puri distinti	Tratti	$Imm(u)$					
		Quad. convesso	Quad. non convesso	Non quad.	Non quad. lineare	Non quad. degenerare 1	Non quad. degenerare 2
4	Segmento	Degenerare	No	No	No	No	No
4	2 segmenti	Degenerare	No	No	No	No	No
4	3 segmenti	Sì	No	No	No	No	No
4	4 segmenti	No	No	Sì	Sì	No	No
4	3 segmenti, 2 discontinuità	No	No	Sì	Sì	No	No
4	3 segmenti, parabola, 2 discontinuità	No	No	Sì	No	No	No
4	2 segmenti, 2 discontinuità	No	No	No	No	Sì	No
4	2 segmenti, parabola, 2 discontinuità	No	No	No	No	Sì	No

## Capitolo 3

# La curva di Pareto nei giochi $n \times n$

### Premesse

Nel capitolo precedente è stato mostrato come il calcolo *esatto* della Curva di Pareto in un gioco a 2 giocatori in forma strategica, 2 mosse ciascuno sia sempre possibile e *facile*. Vogliamo estendere ora il risultato ottenuto al caso generico di bimatrice  $n \times n$ , domandandoci soprattutto quanto sia difficile in termini di complessità temporale. I risultati presenti nella letteratura scientifica ci dicono che in generale (per funzioni obiettivo non convesse) il calcolo della Curva di Pareto richiede un tempo computazionale esponenziale [3, 13, 27]. Nello specifico, le strade già percorse sono diverse: focalizzarsi sull'approssimare la Curva di Pareto mediante rappresentazioni concise [16, 20, 27], fornire risultati per specifici casi in cui gli obiettivi sono convessi [14] e sviluppare approcci evolutivi per generare campioni approssimativamente Pareto efficiente. [8].

### 3.1 Divide et impera

Avendo ben chiaro che il nostro obiettivo è ottenere una rappresentazione esatta della Curva di Pareto con un costo computazionale ragionevole, iniziamo la nostra analisi ricordando un importante risultato ottenuto in [15]:

**Teorema** In un gioco rappresentato dalla bimatrice  $N$  di dimensione  $n \times n$  se un dato payoff è Pareto dominato, allora esiste un payoff appartenente ad una sottobimatrice  $U$  di  $N$ , di dimensione  $2 \times 2$ , che lo domina.<sup>1</sup>

Il teorema ci da importanti informazioni su  $P(N)$ : anzichè considerare la bimatrice  $n \times n$  nella sua interezza, è possibile restringere l'analisi a tutte le sue sottobimatrici

---

<sup>1</sup>Si considera il caso in cui  $N$  sia quadrata per semplicità: il teorema vale anche per una generica bimatrice di dimensione  $n_1 \times n_2$ .

quadrate  $U$  di dimensione 2 ed in particolare a  $P(U)$ . Infatti la Curva di Pareto di  $N$  sarà composta unicamente da punti appartenenti a  $P(U)$  per qualche sottobmatrice  $U$ . Come accade spesso in molti problemi dell'informatica, anche in questo caso un approccio *divide et impera* risulta efficace.

Avendo già mostrato come calcolare  $P(N)$  nel caso base  $n = 2$ , nella nostra situazione il principale ostacolo risiede nella procedura di *ricombinazione* dei risultati ottenuti nel caso base. In altre parole dobbiamo sviluppare un metodo per poter confrontare  $\binom{n}{2}^2$  insiemi  $P(U)$ , selezionando soltanto i punti che non risultano Pareto dominati da nessun altro.

L'Algoritmo 3.1 fornisce una panoramica ad alto livello sugli step da intraprendere per il calcolo di  $P(N)$  attraverso la funzione  $computeParetoCurve(N)$ , mentre l'Algoritmo 3.2 fornisce una panoramica della struttura della funzione di supporto  $subtract(P(U), P(V))$ .

- Funzione  $computeParetoCurve(N)$ :

Inanzitutto, per ognuna delle  $\binom{n}{2}^2$  sottobmatrici  $U$  di dimensione 2, viene calcolato  $P(U)$ . Una volta che l'insieme  $baseParetoCurves$  è completamente popolato, ogni Curva di Pareto  $P(U)$  viene incrementalmente confrontata con tutte le altre curve  $P(V)$  attraverso la funzione  $subtract(temp, P(V))$  il cui valore di ritorno viene usato per aggiornare la variabile  $temp$  stessa, inizialmente posta uguale ad  $P(U)$ . Una volta esauriti i confronti, la variabile  $temp$  conterrà tutti quei punti appartenenti a  $P(U)$  che non sono Pareto dominati da nessun'altra sottobmatrice, quindi verranno aggiunti alla Curva di Pareto  $P(N)$ .

Terminate le  $\binom{n}{2}^4$  iterazioni, in virtù del teorema sopra esposto, avremo terminato la computazione esatta di  $P(N)$ .

- Funzione  $subtract(P(U), P(V))$ :

Questa funzione rappresenta il fulcro centrale della fase di ricombinazione dei risultati, di cui si accennava prima. Essa confronta le Curve di Pareto di giochi 2x2  $P(U)$  e  $P(V)$  (più in generale  $P(U)$  può essere un sottoinsieme di una Curva di Pareto 2x2) e restituisce la collezione di punti appartenenti a  $temp$  che non sono Pareto dominati da nessun punto appartenente a  $P(V)$ . Per illustrare il metodo alla base della funzione  $subtract()$ , sono necessarie delle considerazioni sulla rappresentazione di una generica  $P(U)$ . Nel capitolo precedente è stata dimostrata la seguente proprietà:

**Proprietà** La Curva di Pareto di una generica bimatrice 2x2 è definita *al più* da quattro tratti, ed in particolare ognuno di questi è descritto da un tratto di parabola, un segmento, od un punto.

**Algoritmo 3.1** Calcolo di  $P(N)$ 


---

```

1: function COMPUTEPARETOCURVE(N)
2:    $baseParetoCurves \leftarrow \emptyset$ 
3:    $P(N) \leftarrow \emptyset$ 
4:   for all  $U \in N$  do
5:      $P(U) \leftarrow compute2x2ParetoCurve(U)$ 
6:      $baseParetoCurves \leftarrow baseParetoCurves \cup P(U)$ 
7:   end for
8:   for all  $P(U) \in baseParetoCurves$  do
9:      $temp \leftarrow P(U)$ 
10:    for all  $P(V) \in baseParetoCurves$  do
11:      if  $P(U) \neq P(V)$  then
12:         $temp \leftarrow subtract(temp, P(V))$ 
13:      end if
14:    end for
15:     $P(N) \leftarrow P(N) \cup temp$ 
16:  end for
17:  return  $P(N)$ 
18: end function

```

---

Inoltre poichè stiamo considerando la Curva di Pareto, cioè la collezione di punti che non sono debolmente Pareto dominati, vale anche:

**Proprietà** Ogni tratto è rappresentabile come  $u_2 = f(u_1)$  con  $u_1 \in [u_{1min}, u_{1max}]$  dove l'intervallo può eventualmente diventare aperto.  $f(u_1)$  è inoltre *strettamente* decrescente.

Nel caso di un segmento o di un punto questa rappresentazione è immediata. Nel caso di un tratto di parabola, per ottenere questa rappresentazione si può procedere partendo dalla forma parametrica (2.10) derivando (come nell'analogo teorema del capitolo 2) un'unica equazione nelle componenti cartesiane, nel nostro caso  $u_1$  e  $u_2$ . A questo punto la si può risolvere rispetto ad  $u_2$ : essendo un'equazione di secondo grado in  $u_1$  e  $u_2$ , esisteranno due soluzioni esatte  $u_2 = f(u_1)$  calcolabili in tempo costante, di cui una è quella a cui siamo interessati. È bene ricordare anche che in una data  $P(U)$  gli intervalli di definizione dei tratti che la compongono, non si sovrappongono mai, ma al più possono avere un estremo in comune.

Fatte queste premesse, ci troviamo nel caso di confrontare un generico tratto di  $P(U)$ , definito da  $\bar{f}(u_1)$  con  $u_1 \in [\bar{u}_{1min}, \bar{u}_{1max}]$  (il fatto che l'intervallo sia aperto o chiuso non è rilevante) con uno appartenente a  $P(V)$  definito da  $\tilde{f}(u_1)$  con  $u_1 \in [\tilde{u}_{1min}, \tilde{u}_{1max}]$ . L'idea di base è la stessa utilizzata per derivare tutte le forme possibili di una generica  $P(U)$  a partire dai payoffs puri. A differenza di quella situazione però, il potenziale dominante  $\tilde{f}(u_1)$  è definito su un intervallo, e non in un unico punto come accadeva per un singolo vettore di payoffs puri. La soluzione non è così semplice come possa sembrare: sebbene si tratti di geometria elementare, il numero di casi da considerare è significativo. La presenza di 2 intervalli arbitrari, ci impone di considerare tutte le possibili sovrapposizioni tra questi. L'Algoritmo 3.2 fornisce una panoramica della struttura della funzione *subtract*, in particolare sulla gestione delle casistiche per il confronto di un tratto di  $P(U)$  ed uno di  $P(V)$ , rispetto ai

**Algoritmo 3.2** Struttura di *subtract* ( $P(U), P(V)$ )

---

```

1: function SUBSTRACT(P(U),P(V))
2:   for all  $\bar{f}(u_1) \in P(U)$  do
3:     for all  $\tilde{f}(u_1) \in P(V)$  do
4:       if  $\tilde{u}_{1max} \geq \bar{u}_{1min}$  then
5:         if  $\tilde{u}_{1min} \geq \bar{u}_{1max}$  then
6:           ...
7:         else
8:           if  $\tilde{u}_{1min} > \bar{u}_{1min}$  then
9:             if  $\tilde{u}_{1max} \geq \bar{u}_{1max}$  then
10:              ...
11:            else
12:              ...
13:            end if
14:          else
15:            if  $\tilde{u}_{1max} \geq \bar{u}_{1max}$  then
16:              ...
17:            else
18:              ...
19:            end if
20:          end if
21:        end if
22:      else
23:        end if
24:      end for
25:    end for
26:  end function

```

---

relativi intervalli.

Il primo test effettuato è

$$\tilde{u}_{1max} \geq \bar{u}_{1min}$$

Se questa condizione non fosse verificata il tratto potenzialmente dominante  $\tilde{f}(u_1)$  sarebbe completamente a sinistra (avendo  $u_1$  come ascissa) di  $\bar{f}(u_1)$ , quindi per la definizione di Pareto dominanza potremmo evitare qualsiasi controllo e mantenere integralmente  $\bar{f}(u_1)$ .

Viceversa, se fosse verificata esisterebbe almeno un punto  $\tilde{f}(u_1)$  che potrebbe dominare dei tratti di  $\bar{f}(u_1)$ . Successivamente vengono differenziati cinque casi che presentano proprietà differenti e che coprono la totalità delle eventualità possibili. Nella Tabella 3.1 vengono riportate le condizioni associate ad ogni caso, che risultano rappresentative. Il caso 0 identifica la situazione in cui, avendo posto  $u_1$  come ascissa,  $\tilde{f}(u_1)$  è posto a destra a di  $\bar{f}(u_1)$  e i relativi intervalli di definizione si sovrappongono al massimo in un punto. Questo caso è particolarmente rilevante perchè la sua trattazione è completamente indipendente dai tipi di tratti presi in considerazione (parabola, segmenti o punti). Inoltre, fatto salvo il punto  $\bar{u}_{1max}$  non è possibile che ci siano intersezioni tra  $\tilde{f}(u_1)$  ed  $\bar{f}(u_1)$ . Per tutti gli altri casi invece è necessario tenere in considerazione la tipologia di  $\tilde{f}(u_1)$  ed  $\bar{f}(u_1)$  ed è possibile che esistano una o più interesezioni tra di essi. A tal proposito, considerata la tipologia di tratti possibili vale:

**Proprietà** Presi 2 tratti appartenenti alle Curve di Pareto di due generiche

Tabella 3.1: Casistica sovrapposizione intervalli di  $subtract(P(U), P(V))$ .

Tipologia	Condizioni
Caso 0	$\tilde{u}_{1min} \geq \bar{u}_{1max}$
Caso 1	$\tilde{u}_{1min} > \bar{u}_{1min}$ $\tilde{u}_{1max} \geq \bar{u}_{1max}$
Caso 2	$\tilde{u}_{1min} > \bar{u}_{1min}$ $\tilde{u}_{1max} < \bar{u}_{1max}$
Caso 3	$\tilde{u}_{1min} \leq \bar{u}_{1min}$ $\tilde{u}_{1max} \geq \bar{u}_{1max}$
Caso 4	$\tilde{u}_{1min} \leq \bar{u}_{1min}$ $\tilde{u}_{1max} < \bar{u}_{1max}$

bimatrici  $2 \times 2$ , uno ciascuna, questi s'intersecano al più in due punti.

Questo può avvenire in un confronto tra 2 tratti di parabola, o tra un tratto di parabola ed un segmento. Il caso in cui i tratti siano completamente sovrapposti non è rilevante, in quanto la relazione di Pareto dominanza non è riflessiva. Considerando le tipologie di tratti possibili, nella funzione  $subtract$  può essere necessario calcolare le intersezioni tra due rette, due parabole od una retta ed una parabola. È noto che se esistono soluzioni reali delle equazioni che descrivono queste intersezioni, sono calcolabili (almeno simbolicamente, nel caso di soluzioni irrazionali) esattamente in tempo costante [4].

Fornita la struttura generale della funzione  $subtract$  ed avendo richiamato le principali proprietà sulle quali è costruita, non continuiamo nell'illustrazione in dettaglio di tale funzione, in quanto una volta stabilito lo scheletro generale, la definizione di più di 27 casistiche di confronto geometrico tra i tratti e gli intervalli possibili, costituisce a nostro avviso una technicality di geometria elementare che esula dagli scopi di questo documento. Preferiamo dunque fornire gli *hints* fondamentali che permettano al lettore di immaginare come questi possano essere strutturati, e che ci permettano in seguito di compiere un'analisi di complessità temporale dell'algoritmo.

Come già accennato l'idea con la quale si effettuano i confronti è simile a quella utilizzate per definire le forme delle Curve di Pareto delle bimatrici  $2 \times 2$  a partire dai payoffs puri Pareto efficienti. Nel caso 0 ad esempio, si utilizzerebbe come riferimento per sezionare (eventualmente)  $\bar{f}(u_1)$ , la retta  $u_2 = \tilde{f}(\tilde{u}_{1min})$ . Nel caso 2, senza intersezioni, le rette  $u_2 = \tilde{f}(\tilde{u}_{1min})$  ed  $u_1 = \tilde{u}_{1max}$  e così via. Ci domandiamo ora qual'è il numero massimo di sottointervalli di  $[\bar{u}_{1min}, \bar{u}_{1max}]$  nei quali  $\bar{f}(u_1)$  non risulta Pareto dominata, dopo aver effettuato un confronto. Prendendo l'appena citato caso 2 come riferimento, sarà già evidente al lettore come, senza intersezioni

tra  $\bar{f}(u_1)$  e  $\tilde{f}(u_1)$ , al termine del confronto  $\bar{f}(u_1)$  potrà essere definita al massimo su 2 sottointervalli di quello originale. Anche nell'eventualità che ci fosse una sola intersezione, dopo una breve riflessione sarà chiaro al lettore come potrebbero generarsi al massimo 2 sottointervalli. Se ce ne fossero 2 invece, e nello specifico se entrambe le funzioni fossere curve quadratiche, potrebbero generarsi al massimo 3 sottointervalli. Questa considerazione risulta necessaria per enunciare il seguente:

**Teorema** *subtract* ( $P(U), P(V)$ ) restituisce un insieme di «pezzi», con cardinalità inferiore a 10.

**Dimostrazione** Stiamo considerando il caso in cui *subtract* venga invocata passando come argomenti 2 Curve di pareto  $2 \times 2$ , ognuna delle quali può essere definita al massimo da quattro pezzi. Sappiamo inoltre che al termine di un confronto tra 2 pezzi, possono risultare non Pareto dominati 3 sottointervalli del pezzo di  $P(U)$  che stavamo considerando, se e solo se entrambi i pezzi coinvolti nel confronto sono curve quadratiche. In tutti gli altri casi il massimo numero di sottointervalli che possono venirsi a creare è 2. Infine, gli intervalli su cui sono definiti i tratti di una qualsiasi  $2 \times 2$  solo al più adiacenti, e mai sovrapposti. Questo significa che, anche considerando eventuali sovrapposizioni degli intervalli, nel caso pessimo potremmo avere un confronto di 2 tratti curvilinei che genera 3 pezzi, e rimanenti 3 confronti tra segmenti che generano 2 pezzi ciascuno, per un totale di massimo 9 pezzi.

## 3.2 Complessità temporale

Il teorema enunciato, fornisce un upperbound *estremamente* prudente. Infatti l'analisi effettuata per fornire questo tetto massimo, considera singolarmente ogni confronto valutandone il caso pessimo. Tuttavia nel capitolo precedente è stato mostrato come possano esserci al massimo 2 discontinuità in una Curva di Pareto  $2 \times 2$ , cioè in altre parole, che considerando curve di 4 pezzi ciascuna, il caso pessimo su ognuno dei 4 pezzi non si può comunque verificare. Ancora più prudente sarà l'enunciato del prossimo lemma, che prima di riportare, ha bisogno di una proposizione come premessa:

**Proposizione** L'utilizzo di *subtract* in maniera incrementale, come quello fatto nell'algoritmo 3.1 restituisce come risultato l'insieme di tutti e soli punti di  $P(U)$  che non sono Pareto dominati da nessuna delle  $P(V)$  passate come argomento.

**Lemma** Il numero di pezzi di  $P(U)$  che non sono Pareto dominati da  $k$  Curve di Pareto  $2 \times 2$  differenti da essa, è al più  $9k$ .

Oltre a partire da un ipotesi molto conservativa, anche per questo lemma si potrebbero fare le stesse considerazioni fatte per il teorema precedente: ad esempio anche ammettendo che alla prima chiamata  $subtract(temp, P(V))$  abbia ritornato 9 pezzi differenti, alla seconda iterazione  $subtract(temp, P(V))$  non ne ritornerà 18, in virtù dei motivi sopra esposti. Questi risultati quindi, seppur molto laschi, sono tuttavia validi e danno comunque delle informazioni importanti per un approccio preliminare ad un argomento non ancora trattato in letteratura. Riprenderemo questi risultati, dopo aver svolto la (semplice) analisi di complessità temporale dell'algoritmo 3.1

**Teorema** Il calcolo della Curva di Pareto di un gioco rappresentato una bimatrice quadrata  $N$  di ordine  $n$  appartiene alla classe  $\mathcal{FP}$ , ed ha una complessità temporale  $T(n) = \mathcal{O}(n^{12})$

**Dimostrazione** La prima parte dell'algoritmo calcola la Curva di Pareto di ogni sottobimatrice di  $N$  quadrata di ordine 2, cioè di  $\binom{n}{2}^2$  bimatrici 2x2. Nella seconda invece viene svolta la parte relativa ai confronti tra esse. Al termine di ogni iterazione del ciclo più esterno (riga 8 dell'algoritmo 3.1) nella variabile  $temp$  sono contenuti appartenenti a  $P(U)$  che appartengono anche a  $P(N)$ . Il numero di queste iterazioni è esattamente il numero di sottobimatrici  $2 \times 2$ , cioè  $\binom{n}{2}^2$ . Domandiamoci ora quanto costa, per ogni  $P(U)$  confrontarla tutte le altre. Chiamando  $T^1(k)$  la complessità complessiva computazionale dalla riga 10 alla 14, con  $k$  il generico numero di elementi in  $baseParetoCurves$  abbiamo nel caso pessimo:

$$T^1(k) = 1 + 4 \times 4 + 4 \times 9 + 4 \times 18 + \dots + 4 \times 9 \cdot (k - 2) \quad (3.1)$$

Infatti nella prima iterazione  $subtract$  nel caso pessimo confronterà ognuno dei 4 pezzi di  $P(U)$  con i 4 di  $P(V)$ , alla seconda ognuno dei 9 della variabile  $temp$  con 4 della successiva  $P(V)$  e così via per tutte le  $k - 1$  bimatrici differenti da  $P(U)$ . Il primo termine unitario rappresenta l'iterazione in cui  $P(U) = P(V)$ . Riscriviamo l'equazione 3.1 in maniera che risulti più leggibile

$$T^1(k) = 1 + 16 + 4 \times 9 \times (1 + 2 + \dots + k - 2)$$

Cioè:

$$T^1(k) = \mathcal{O}(1) + 36 \cdot \frac{(k - 2) \cdot (k - 1)}{2} = \mathcal{O}(k^2)$$

Poichè nel nostro caso  $baseParetoCurves$  possiede  $\binom{n}{2}^2$  elementi, abbiamo

quindi una complessità totale per l'algoritmo del calcolo di  $P(N)$  di:

$$T(n) = \binom{n}{2}^2 + \binom{n}{2}^2 \cdot \mathcal{O} \left( \binom{n}{2}^4 \right) + \binom{n}{2}^2$$

Quindi:

$$T(n) = \mathcal{O} \left( \binom{n}{2}^2 \right) + \mathcal{O} \left( \binom{n}{2}^6 \right) = \mathcal{O} (n^{12})$$

Che dimostra il teorema.

Il risultato è importante: abbiamo dimostrato che per un *qualsiasi* gioco a 2 giocatori rappresentabile in forma strategica, il calcolo *esatto* di *tutti* i punti Pareto efficienti richiede un tempo polinomiale rispetto alle dimensioni dell'istanza, cioè che il problema appartiene a  $\mathcal{FP}$ . Come nel caso  $2 \times 2$ , anche generalizzando ad un numero arbitrario di mosse, non è rilevante se gli elementi della bimatrice siano razionali o irrazionali: in ambo le casistiche è possibile che le soluzioni delle equazioni utilizzate nella procedura per il calcolo della Curva di Pareto siano irrazionali. In questo caso non avrebbe senso parlare di complessità temporale dell'algoritmo in  $\mathcal{FP}$  o  $\mathcal{FNP}$  cercando una rappresentazione numerica dei risultati. Da un punto di vista simbolico però, abbiamo dimostrato che l'algoritmo non effettua nessuna approssimazione, in quanto per tutte le equazioni coinvolte sono noti metodi risolutivi esatti. Pertanto possiamo effettuare il calcolo della curva in maniera esatta sfruttando una rappresentazione simbolica, e poi scelta una precisione arbitraria (ma finita) rappresentarla anche numericamente.

Concludendo, contrariamente al caso generale in cui il calcolo della Curva di Pareto non può essere svolto in tempo polinomiale [3, 13, 27], per questa specifica classe di giochi è possibile. Oltre ad avere effettuato un'analisi della complessità temporale dell'algoritmo di calcolo di  $P(N)$  possiamo enunciare anche:

**Corollario** La curva di Pareto di un gioco rappresentato una bimatrice quadrata  $N$  di ordine  $n$  è composta da meno di  $9n^{12}$  pezzi.

### 3.3 Analisi sperimentale

Abbiamo effettuato un'analisi sperimentale dell'algoritmo su due bimatrici  $10 \times 10$ , e due bimatrici  $5 \times 5$  costituite da elementi generati casualmente con una distribuzione uniforme tra 0 e 1. Di seguito vengono riportate le bimatrici ed i relativi grafici. I vincoli derivati precedentemente sul numero di pezzi della curva, è ampiamente rispettato.

0.815	0.098	0.158	0.142	0.656	0.758	0.706	0.823	0.439	0.490
0.906	0.278	0.971	0.422	0.036	0.743	0.032	0.695	0.382	0.446
0.127	0.547	0.957	0.916	0.849	0.392	0.277	0.317	0.766	0.646
0.913	0.958	0.485	0.792	0.934	0.655	0.046	0.950	0.795	0.709
0.632	0.965	0.800	0.959	0.679	0.171	0.097	0.034	0.187	0.755

Table 3.2: Bimatrice casuale  $5 \times 5$ .

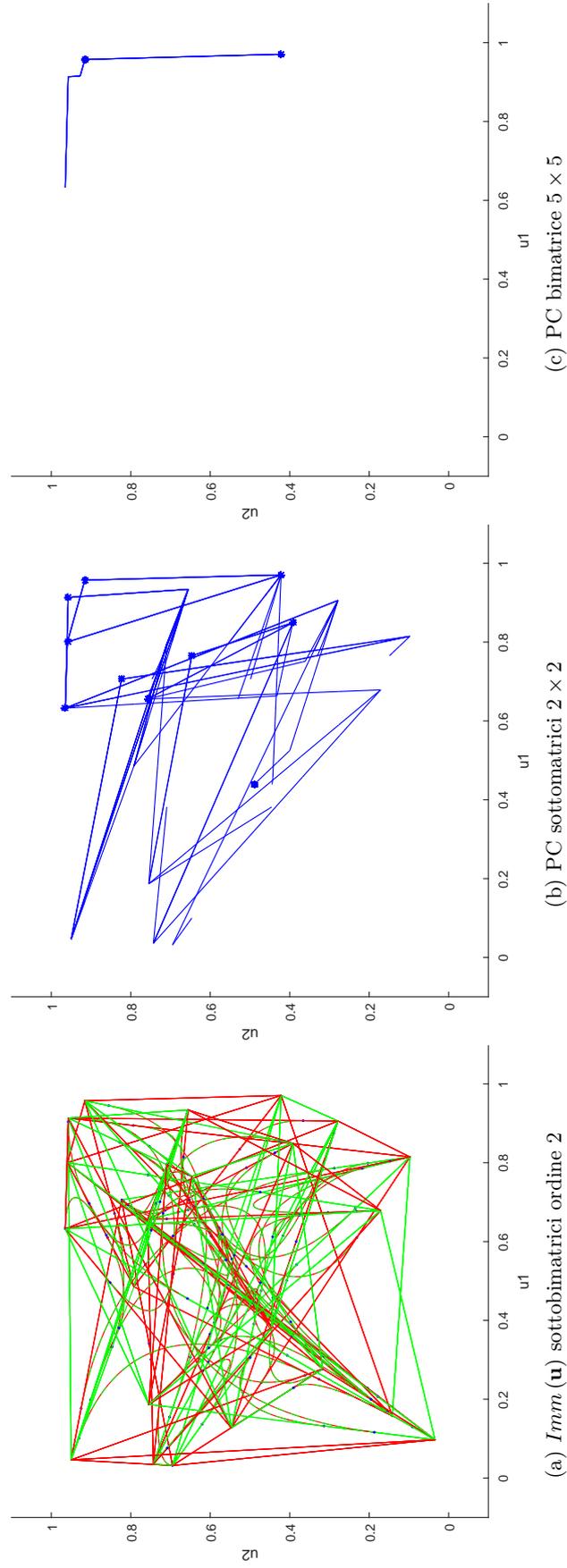


Figura 3.1: Curva di Pareto della bimatrice  $5 \times 5$  in Tabella 3.2.

0.276	0.498	0.751	0.959	0.841	0.350	0.352	0.286	0.076	0.130
0.680	0.960	0.255	0.547	0.254	0.197	0.831	0.757	0.054	0.569
0.655	0.340	0.506	0.139	0.814	0.251	0.585	0.754	0.531	0.469
0.163	0.585	0.699	0.149	0.244	0.616	0.550	0.380	0.779	0.012
0.119	0.224	0.891	0.258	0.929	0.473	0.917	0.568	0.934	0.337

Table 3.3: Bimatrice casuale  $5 \times 5$ .

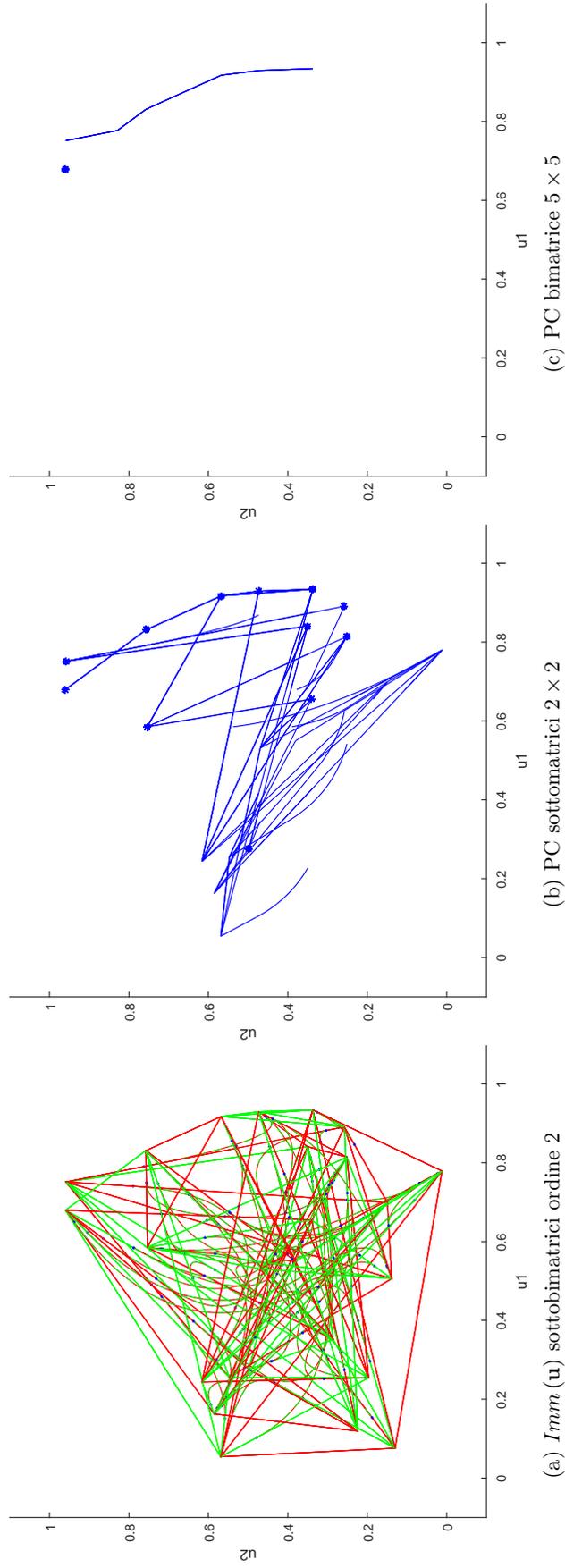


Figura 3.2: Curva di Pareto della bimatrice  $5 \times 5$  in Tabella 3.3.

0.644	0.208	0.311	0.595	0.086	0.963	0.038	0.107	0.031	0.183	0.060	0.660	0.173	0.292	0.952	0.301	0.033	0.645	0.252	0.906
0.379	0.301	0.923	0.262	0.262	0.547	0.885	0.654	0.744	0.240	0.682	0.519	0.391	0.432	0.920	0.701	0.561	0.376	0.290	0.880
0.812	0.471	0.430	0.603	0.801	0.521	0.913	0.494	0.500	0.887	0.042	0.973	0.831	0.015	0.053	0.666	0.882	0.191	0.617	0.818
0.533	0.230	0.185	0.711	0.029	0.232	0.796	0.779	0.480	0.029	0.071	0.649	0.803	0.984	0.738	0.539	0.669	0.428	0.265	0.261
0.351	0.844	0.905	0.222	0.929	0.489	0.099	0.715	0.905	0.490	0.522	0.800	0.060	0.167	0.269	0.698	0.190	0.482	0.824	0.594
0.939	0.195	0.980	0.117	0.730	0.624	0.262	0.904	0.610	0.168	0.097	0.454	0.399	0.106	0.423	0.667	0.369	0.121	0.983	0.023
0.876	0.226	0.439	0.297	0.489	0.679	0.335	0.891	0.618	0.979	0.818	0.432	0.527	0.372	0.548	0.178	0.461	0.590	0.730	0.425
0.550	0.171	0.111	0.319	0.579	0.396	0.680	0.334	0.859	0.713	0.818	0.825	0.417	0.198	0.943	0.128	0.982	0.226	0.344	0.313
0.622	0.228	0.258	0.424	0.237	0.367	0.137	0.699	0.805	0.500	0.722	0.083	0.657	0.490	0.418	0.999	0.156	0.385	0.584	0.161
0.587	0.436	0.409	0.508	0.459	0.988	0.721	0.198	0.577	0.471	0.150	0.133	0.628	0.339	0.983	0.171	0.856	0.583	0.108	0.179

Table 3.4: Bimatrice casuale  $10 \times 10$ .

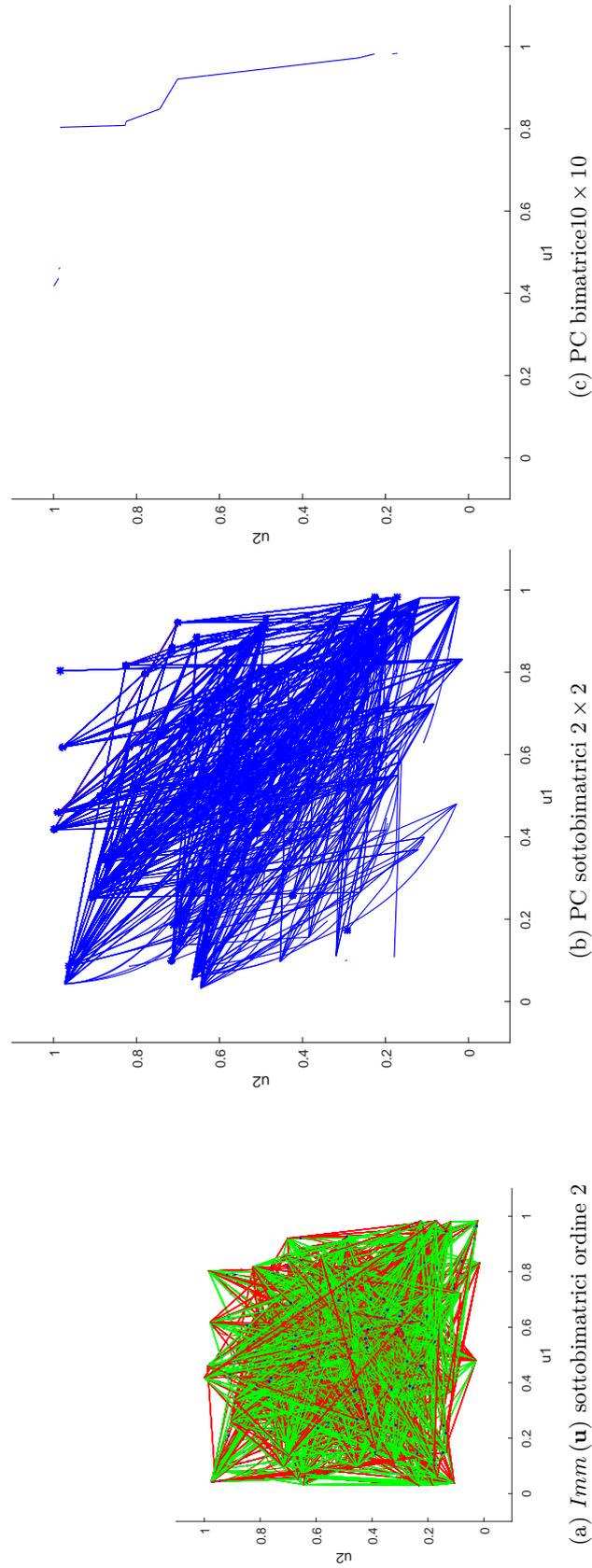


Figura 3.3: Curva di Pareto della bimatrice  $10 \times 10$  in Tabella 3.4.

0.815	0.158	0.656	0.706	0.439	0.276	0.751	0.841	0.352	0.076	0.162	0.451	0.107	0.431	0.853	0.417	0.780	0.235	0.547	0.929
0.906	0.971	0.036	0.032	0.382	0.680	0.255	0.254	0.831	0.054	0.794	0.084	0.962	0.911	0.622	0.050	0.390	0.353	0.296	0.776
0.127	0.957	0.849	0.277	0.766	0.655	0.506	0.814	0.585	0.531	0.311	0.229	0.005	0.182	0.351	0.903	0.242	0.821	0.745	0.487
0.913	0.485	0.934	0.046	0.795	0.163	0.699	0.244	0.550	0.779	0.529	0.913	0.775	0.264	0.513	0.945	0.404	0.015	0.189	0.436
0.632	0.800	0.679	0.097	0.187	0.119	0.891	0.929	0.917	0.934	0.166	0.152	0.817	0.146	0.402	0.491	0.096	0.043	0.687	0.447
0.098	0.142	0.758	0.823	0.490	0.498	0.959	0.350	0.286	0.130	0.602	0.826	0.869	0.136	0.076	0.489	0.132	0.169	0.184	0.306
0.278	0.422	0.743	0.695	0.446	0.960	0.547	0.197	0.757	0.569	0.263	0.538	0.084	0.869	0.240	0.338	0.942	0.649	0.368	0.509
0.547	0.916	0.392	0.317	0.646	0.340	0.139	0.251	0.754	0.469	0.654	0.996	0.400	0.580	0.123	0.900	0.956	0.732	0.626	0.511
0.958	0.792	0.655	0.950	0.709	0.585	0.149	0.616	0.380	0.012	0.689	0.078	0.260	0.550	0.184	0.369	0.575	0.648	0.780	0.818
0.965	0.959	0.171	0.034	0.755	0.224	0.258	0.473	0.568	0.337	0.748	0.443	0.800	0.145	0.240	0.111	0.060	0.451	0.081	0.795

Table 3.5: Bimatrice casuale  $10 \times 10$ .

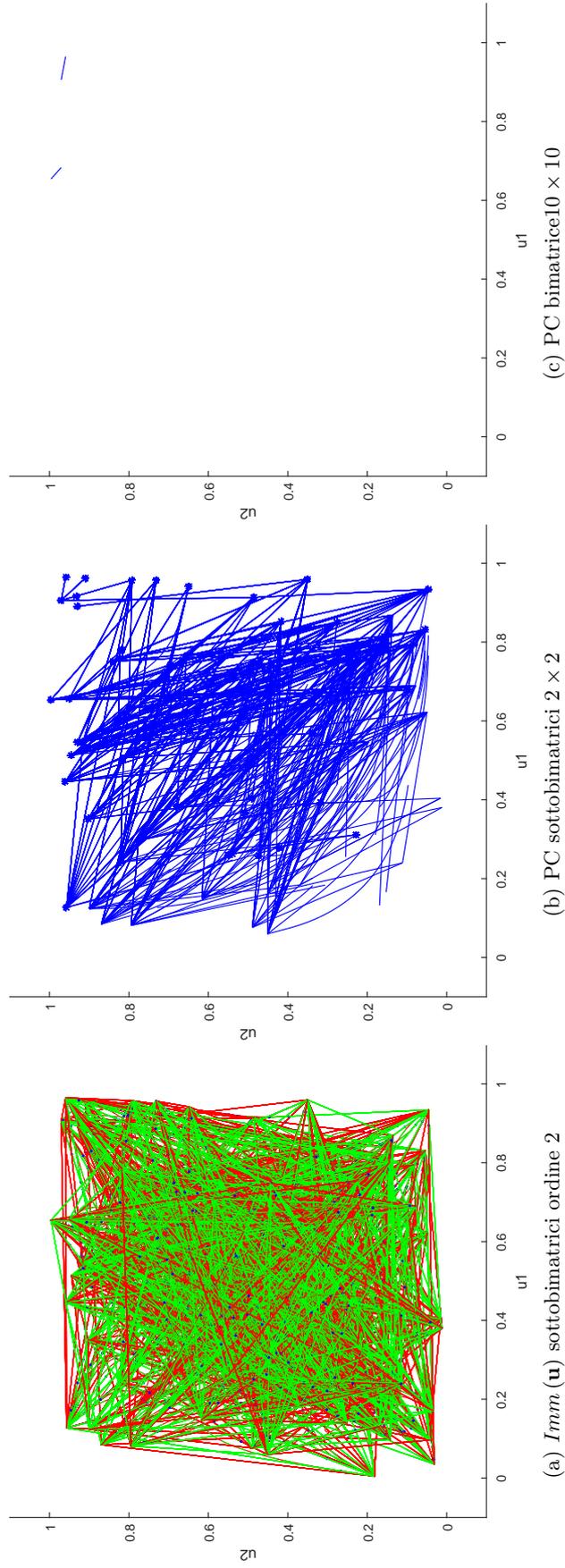


Figura 3.4: Curva di Pareto della bimatrice  $10 \times 10$  in Tabella 3.5.

### 3.4 Algoritmi per concetti di soluzione basati sulla Curva di Pareto

Utilizziamo il risultato ottenuto nella sezione precedente per fornire degli algoritmi che calcolano solutions concepts basati sulla Pareto efficienza.

**Teorema** La *Utilitarian bargaining solution* (UBS) [32] di un gioco bimatrice  $n \times n$  può essere calcolata in tempo polinomiale.

**Dimostrazione** Abbiamo dimostrato come sia possibile calcolare la Curva di Pareto per tale classe di giochi in tempo polinomiale rispetto ad  $n$ , e che tale curva è composta da un numero polinomiale (sempre rispetto ad  $n$ ) di pezzi. Per calcolare UBS, è necessario calcolare per ogni pezzo che costituisce la Curva di Pareto il valore di  $u_1$  che massimizza  $u_1 + u_2^*(u_1)$  sotto il vincolo  $u_1 \in [(u_2^*)^{-1}(u_2^{max}), u_1^{max}]$ . Questo è un problema di ottimizzazione di una funzione quadratica in una sola variabile, e può essere risolto esattamente in tempo costante. Una volta trovato il valore ottimo per ogni pezzo, è sufficiente selezionare tra tutti, il massimo. Poichè abbiamo un numero polinomiale in  $n$  di pezzi, l'algoritmo termina in tempo polinomiale, ed in particolare UBS può essere calcolata in  $\mathcal{O}(n^{12})$ .

**Teorema** La *Nash bargaining solution* (NBS) [24] di un gioco bimatrice  $n \times n$  può essere calcolata in tempo polinomiale.

**Dimostrazione** Per calcolare NBS, è necessario calcolare per ogni pezzo che costituisce la Curva di Pareto il valore di  $u_1$  che massimizza  $u_1 * u_2^*(u_1)$  sotto il vincolo  $u_1 \in [(u_2^*)^{-1}(u_2^{max}), u_1^{max}]$ . Questo è un problema di ottimizzazione di una funzione cubica in una sola variabile, e può essere risolto esattamente in tempo costante. Una volta trovato il valore ottimo per ogni pezzo, è sufficiente selezionare tra tutti, il massimo. Poichè abbiamo un numero polinomiale in  $n$  di pezzi, l'algoritmo termina in tempo polinomiale, ed in particolare NBS può essere calcolata in  $\mathcal{O}(n^{12})$ .

**Teorema** La *Kalai-Smorodinsky bargaining solution* (KSBS) [18] di un gioco bimatrice  $n \times n$  può essere calcolata in tempo polinomiale (se esiste).

**Dimostrazione** Per calcolare NBS, è necessario calcolare per ogni pezzo che costituisce la Curva di Pareto il valore di  $u_1$  tale che  $\frac{u_1}{u_2^*(u_1)} = \frac{u_1^{max}}{u_2^{max}}$  sotto il vincolo  $u_1 \in [(u_2^*)^{-1}(u_2^{max}), u_1^{max}]$ . Stiamo risolvendo cioè l'equazione  $u_1 = u_2^*(u_1) \frac{u_1^{max}}{u_2^{max}}$  che è quadratica in una variabile, e può essere risolta esattamente in tempo costante. Quindi, se esiste, KSBS può essere calcolata in  $\mathcal{O}(n^{12})$ .

# Conclusioni e sviluppi futuri

## Conclusioni

Il problema di ricerca di soluzioni Pareto efficienti in un generico problema di ottimizzazione multiobiettivo, è computazionalmente intrattabile. L'approccio prevalente in letteratura è quello di cercare soluzioni approssimate, attraverso diverse metodologie. Non essendo presenti risultati in letteratura, abbiamo focalizzato l'attenzione sui problemi di giochi non cooperativi. In particolare, partendo da quanto evidenziato in [15], abbiamo ristretto ulteriormente il campo ai giochi in forma strategica a 2 giocatori. Abbiamo considerato inizialmente i giochi bimatrice  $2 \times 2$ , e sulla base di considerazioni di tipo geometrico, abbiamo caratterizzato la frontiera dell'insieme dei possibili outcome del gioco, mostrando che è definita a tratti da segmenti e al più da un tratto di parabola.

Sulla base di questo abbiamo successivamente come l'insieme dei possibili outcome di giochi di questo tipo abbia uno shape canonico, suddivisibile principalmente in due categorie: quadrilaterale e non quadrilaterale. Considerando tutte le degenerazioni possibili, l'insieme dei possibili outcome di un generico gioco  $2 \times 2$  può avere 9 shapes differenti. Con questa caratterizzazione è stato possibile poi ricercare la Curva di Pareto di tali giochi, affrontando il problema non più direttamente in modo analitico come un problema di ottimizzazione, ma utilizzando i risultati precedentemente ottenuti come base di partenza. Differenziando ogni caso sulla base del numero di payoff puri distinti presenti, abbiamo effettuato una classificazione esaustiva della Curva di Pareto per giochi  $2 \times 2$ , costituita da 27 casi differenti. Il calcolo della Curva di Pareto, e della frontiera dell'insieme degli outcome è effettuabile facilmente, in maniera esatta. Infine abbiamo esteso il risultato al caso bimatrice con un numero generico di mosse: ogni punto Pareto efficiente, è compreso in almeno una sottomatrice quadrata di ordine 2.

Grazie alla semplicità delle equazioni che descrivono la Curva di Pareto di una bimatrice  $2 \times 2$ , è stato possibile confrontare più Curve di Pareto  $2 \times 2$  tra loro, selezionando i punti non dominati in maniera *esatta*. Utilizzando un approccio di forza bruta, considerando *tutte* le sottomatrici  $2 \times 2$  di una generica bimatrice, è possibile calcolare la Curva di Pareto in un tempo che cresce polinomialmente con il numero

delle azioni della bimatrice e che è composta da un numero di tratti polinomiale rispetto al numero delle azioni.

## Sviluppi futuri

Le istanze di bimatrici casuali utilizzate nella nostra analisi per il calcolo sperimentale della Curva di Pareto suggeriscono l'esistenza di vincoli molto più stringenti rispetto a quelli qui derivati. Questo appare ragionevole in quanto lo studio di complessità effettuato non ha tenuto in considerazione tra le altre cose, il fatto che le Curve di Pareto  $2 \times 2$  da confrontare, non sono totalmente indipendenti tra loro, in quanto selezionate da una stessa bimatrice di partenza. Considerazioni di questo tipo, o di carattere geometrico, suggeriscono quindi che ci siano buoni margini di miglioramento per la complessità dell'algoritmo,, utlizzando ad esempio tecniche di *pruning* o di programmazione dinamica.

Un approfondimento possibile potrebbe essere quello di indagare la complessità temporale di algoritmi per la ricerca della Curva di Pareto, per la classe di *polymatrix games*, che presentano proprietà simili ai giochi bimatrice.

Infine ci si può domandare anche se il problema rimane trattabile nel caso di giochi a 3 o più giocatori.

# Bibliografia

- [1] R. J. Aumann. Game theory: Introduction. In *The New Palgrave Dictionary of Economics*, pages 529–558. London: Palgrave Macmillan, 2008.
- [2] Robert J Aumann. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics*, 1(1):67–96, 1974.
- [3] Haris Aziz, Felix Brandt, and Paul Harrenstein. Pareto optimality in coalition formation. *Games and Economic Behavior*, 82:562–581, 2013.
- [4] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. Algorithms in real algebraic geometry. *AMC*, 10:12, 2011.
- [5] Karl Borch. Application of game theory to some problems in automobile insurance. *Astin Bulletin*, 2(02):208–221, 1962.
- [6] Eleonora Braggion, Nicola Gatti, Roberto Lucchetti, and Tuomas Sandholm. Strong nash equilibria and mixed strategies. *arXiv preprint arXiv:1502.05629*, 2015.
- [7] Richard Bevan Braithwaite. Theory of games as a tool for the moral philosopher. an inaugural lecture delivered in cambridge on 2 december 1954. 1955.
- [8] Karl Bringmann, Tobias Friedrich, Frank Neumann, and Markus Wagner. Approximation-guided evolutionary multi-objective optimization. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1198, 2011.
- [9] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Settling the complexity of computing two-player nash equilibria. *Journal of the ACM (JACM)*, 56(3):14, 2009.
- [10] Antoine-Augustin Cournot. *Recherches sur les principes mathématiques de la théorie des richesses*. Paris: L. Hachette, 1838.
- [11] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.

- 
- [12] Pierre Rémond de Montmort. *Essay d'analyse sur les jeux de hazard*. Paris: J. Quilau, 1713.
- [13] Thomas Demuynck. The computational complexity of rationalizing pareto optimal choice behavior. *Social Choice and Welfare*, 42(3):529–549, 2014.
- [14] Matthias Ehrgott, Lizhen Shao, and Anita Schöbel. An approximation algorithm for convex multi-objective programming problems. *Journal of Global Optimization*, 50(3):397–416, 2011.
- [15] Nicola Gatti, Marco Rocco, and Tuomas Sandholm. On the verification and computation of strong nash equilibrium. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 723–730. International Foundation for Autonomous Agents and Multiagent Systems, 2013.
- [16] Christian Glaßer, Christian Reitwießner, Heinz Schmitz, and Maximilian Witek. Approximability and hardness in multi-objective optimization. In *Programs, Proofs, Processes*, pages 180–189. Springer, 2010.
- [17] Oskar Morgenstern John von Neumann. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.
- [18] Ehud Kalai and Meir Smorodinsky. Other solutions to nash’s bargaining problem. *Econometrica: Journal of the Econometric Society*, pages 513–518, 1975.
- [19] J Dennis Lawrence. *A catalog of special plane curves*. Courier Corporation, 2013.
- [20] Julien Legriel, Colas Le Guernic, Scott Cotton, and Oded Maler. Approximating the pareto front of multi-criteria optimization problems. In *TACAS*, pages 69–83. Springer, 2010.
- [21] John Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.
- [22] John Nash. Two-person cooperative games. *Econometrica: Journal of the Econometric Society*, pages 128–140, 1953.
- [23] John F Nash et al. Equilibrium points in n-person games. *Proc. Nat. Acad. Sci. USA*, 36(1):48–49, 1950.
- [24] John F Nash Jr. The bargaining problem. *Econometrica: Journal of the Econometric Society*, pages 155–162, 1950.

- 
- [25] J v Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.
- [26] Christos H Papadimitriou. *Computational complexity*. John Wiley and Sons Ltd., 2003.
- [27] Christos H Papadimitriou and Mihalis Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 86–92. IEEE, 2000.
- [28] Vilfredo Pareto. *Manuale di economia politica*, volume 13. Societa Editrice, 1906.
- [29] Reinhard Selten. Spieltheoretische behandlung eines oligopolmodells mit nachfragerträchtigkeit: Teil i: Bestimmung des dynamischen preisgleichgewichts. *Zeitschrift für die gesamte Staatswissenschaft/Journal of Institutional and Theoretical Economics*, pages 301–324, 1965.
- [30] Lloyd S Shapley. A value for n-person games. Technical report, DTIC Document, 1952.
- [31] Lloyd S Shapley and Martin Shubik. A method for evaluating the distribution of power in a committee system. *American Political Science Review*, 48(03):787–792, 1954.
- [32] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.
- [33] Martin Shubik. Strategy and market structure: competition, oligopoly, and the theory of games. 1959.
- [34] J Maynard Smith and GR Price. The logic of animal conflict. *Nature*, 246:15, 1973.
- [35] John Maynard Smith. *Evolution and the Theory of Games*. Cambridge university press, 1982.
- [36] Robert Carl Yates. *A Handbook on curves and their properties*. JW Edwards, 1947.
- [37] Ernst Zermelo. Über eine anwendung der mengenlehre auf die theorie des schachspiels. In *Proceedings of the fifth international congress of mathematicians*, volume 2, pages 501–504. II, Cambridge UP, Cambridge, 1913.