Model for multi-agent Optimisation

Ambrose Yim (University of Oxford)

2 May 2018

Introduction

Let there be N agents indexed by $i \in \mathcal{I}$ working together on a project proposal. Each agent has a (finite) set of actions A_i which they could employ to modify particular contents of the project proposal. The state of the entire project proposal sits in the product space of all actions $A = A_1 \times \ldots \times A_N$. We associate to each agent a utility function, $u : \mathcal{I} \times A \to \mathbb{R}$, which we can interpret as the agent's happiness at a given state.

While the objective of an individual agent is to selfishly maximise their own utility, the project only goes forward when all of the agents are 'sufficiently happy': If any one agent is too unhappy, the project cannot proceed.

Firstly, we define the **collective optimisation problem** (the Co-Op) where where all agents sit together, *pool* their information - i.e. their utility functions - and *altruistically* maximise their **collective benefit function** $U(a) = \min_{i \in \mathcal{I}} u_i(a)$, so that no agent is 'left behind' in the name of 'progress' [1]:

$$\max_{a \in A} \min_{i \in \mathcal{I}} u_i(a) = U^*$$

The optimum (optima) of this problem is $U^* = u_{i*}(a^*)$. There are good reasons to choose this philosophy: quite often there is no actual global utility function to optimise over agents. Multi-agent algorithms often optimise the sum of local utility functions which is an arbitrary choice. A bad reason to adopt this philosophy is when there is a sensible definition of global utility function.

Suppose each agent acts selfishly and wants to push the project towards a state that maximises their own utility $\max_{a \in A} u_i(a)$. Then if all agents optimising selfishly can all do better then U^*, then they would not end up in the state a^* that optimises Co-Op. In other words, the condition that selfish agents fail to 'Co-Op' is

$$\max_{a \in A} \min_{i \in \mathcal{I}} u_i(a) < \min_{i \in \mathcal{I}} \max_{a \in A} u_i(a)$$

However there is this famous inequality, well known in game theory and optimisation circles:

$$\max_{a \in A} \min_{i \in \mathcal{I}} u_i(a) \leq \min_{i \in \mathcal{I}} \max_{a \in A} u_i(a)$$

with stringent conditions on $\mathcal{I}$, A and u for equality to hold [2]. So generically, selfish actions cannot lead to the 'Co-op' communist utopia.

    Question: Can agents strike a balance between altruistic and selfish intentions?

Another idea: Ising model. Frustrated spins.

---

[1] *Aside*: if u is a binary function, where an agent gives assent to a project proposal if $u = 1$ and vetoes the proposal if $u = 0$, the Co-Op optimum is simply the condition that $u_i(a^*) = 1 \quad \forall i \in \mathcal{I}$.

[2] There is a theorem in convex optimisation - if $\mathcal{I}$, $\mathcal{A}$ are convex subsets of some field (e.g. rationals, reals) and u is a convex function then equality is ensured.

Model for Collaborative Process

However there is more finesse to this argument. The assumption that a selfish agent can maximise over all states A is unrealistic - that is what each agent wants; whether it can actually do so is another matter entirely, restricted by its restricted influence on the project proposal. In complex problems (like chess) it is also unlikely that it would be computationally efficient to pre-compute u for all states of the system - usually we can only evaluate u *during* the game.

This points towards the temporal nature of real life organisational optimisation problems.

Coalitions and Coalition Operators

Suppose in an optimisation problem we start in some state $a_0 \in A$ and evolves in discrete time steps $a_0 \to a_1 \to \cdots \to a_T$. Let $K = \{C \subseteq \mathcal{I}\}$ be the collection of all subsets of agents and denote $A(C)$ to be the actions available to agents in C. At any moment t a subset of agents C_t come together as a **coalition** to solve the the problem

$$a_t = \arg \max_{a \in A(C_t)} \min_{i \in C_t} u_i(a|a_{t-1}) = \mathcal{C}_t \ a_{t-1}.$$

We call $\mathcal{C} : A \to A$ a **coalition operator**.

N.B. in a coalition, members

- Coordinate their actions (in max operation)
- Share information about their utility function (in min operation)
- The state space they could explore is restricted by its immediate predecessor a_{t-1}
- The coalition updates the system to a new state where the worse off member of the coalition has a greater utility

Points to Explore:

- What are the constraints on the coalitions that we are allowed to form?
- There is scope for the coalition to be more altruistic/more knowledgeable, by minimi;sing u_i over a superset of C_t (i.e. agents have limited actions but better information and less selfish)
- How does the problem restrict the image of coalition operators?

Collaborative Process

Thus in our framework we can define a **collaborative process**, which is prescribed by a tuple of a set of agents, actions associated with each agent, a utility function, an initial state and a *time-ordered* set of coalitions $\langle \mathcal{I}, A, u, a_0, K \rangle$ from which a chain sequence of states is evolved

$$a_0 \xrightarrow{\mathcal{C}_1} a_1 \xrightarrow{\mathcal{C}_2} \ldots \xrightarrow{\mathcal{C}_T} a_T,$$

where

$$a_t = \mathcal{C}_t \mathcal{C}_{t-1} \ldots \mathcal{C}_1 a_0.$$

To abstract the problem in a more succinct manner, we associate to every ordered set of coalitions a **sequence operator** $\mathcal{K} : A \to A$, which is a map from an initial state to a final state of the sequence $\mathcal{K} = \mathcal{C}_T \mathcal{C}_{T-1} \ldots \mathcal{C}_1$. We observe that the set of sequence operators Γ is a *semigroup* finitely generated by coalition operators $\mathcal{C}$[3].

Points to Explore:

---

[3]that is, Γ is a 'group' without identity and inverses, and every member can be obtained by multiplying together some $\mathcal{C}$'s.

- Is there something special about the algebra of Γ?
- How could we exploit the structure of the problem (namely, the utility matrix u) to design a sequence K that maximises $U(a_T)$ with as few time steps T as possible, or make sure $U(a_t)$ always decreases? e.g. select certain members of C_t to carry forward to C_{t+1} under some criteria?
- Is it possible to actually design K that decreases U for a general problem?
- Effect of 'gerrymandering' coalitions.

Utility functions with Different Properties

Independent Preferences vs Independent Payoff

We say each agent has an **independent preference** for their course of action if each agent i acting selfishly maintains has the same *ordering* of actions regardless of the choices of other agents. More precisely,

$$u_i(a_i \mid a'_{-i}) \geq u_i(a'_i \mid a'_{-i}) \quad \forall a, a' \in A, \quad \forall i \in \mathcal{I}.$$

If agents with independent preference were playing a *simultaneous game*, then there is a **(Pareto) dominant strategy solution** $a^* = (a_1^*, \ldots, a_N^*)$ where everyone would take their most favoured action $a_i^* \in A_i$: it would be the best thing that anyone can do regardless of how other agents play. Yet in Co-Op a^* will not be the generic solution. Bear in mind having an independent preference does not exclude the case

$$\exists a'_{-i} \quad \text{s.t.} \quad u_i(a_i^* \mid a_{-i}) \geq u_i(a_i^* \mid a'_{-i})$$

where an agent's best payoff can be reduced or increased by actions of other agents. Thus in the Co-Op scenario agents might have to relinquish their most preferred action α_i^* to lift the payoff of the least fortunate agent.

> This demonstrates how Co-Op might be a bad philosophy if too many agents had to compromise. However, smaller and 'coalitions' might lead to fewer compromises

We note that agents having **independent payoffs** i.e. the actual *value* of the payoff is independent of the actions of other agents is a stronger statement

$$u_i(a_i \mid a_{-i}) = u_i(a_i \mid a'_{-i}) \quad \forall \, a, a' \in A, \quad \forall i \in \mathcal{I}.$$

Independent payoff implies independent preference, but not the other way round. These two conditions have distinct pathologies. In the case of independent payoffs, $a^* = (a_1^*, \ldots, a_N^*)$ is not only the dominant strategy solution in a simultaneous game, it is also admitted as a solution of Co-Op; for if agents agree on a^* the utility of the least fortunate agent cannot be improved by knocking other agents off their best actions. However, since Co-Op is insensitive to the plight of agents in better situations, there might not be a mechanism to enforce a^* if deviating away from a^* does not lower the utility of the worse performing agent.

Biased Global Optimisation

Suppose there is an overarching objective $V(a)$ that nominally all agents want to jointly optimise. Then we can model the individual agent payoffs as $u_i(a) = V(a) + \epsilon\mu_i(a)$, where μ_i represents an individual **bias** in evaluating a states of affairs; if we normalise V, $\mu_i \in [0,1]$, then ϵ is a measure of the relative strength of individual biases.

Zero-Sum Game

In a **zero-sum game**,

$$\sum_{i \in \mathcal{I}} u_i(a) = 0 \quad \forall \, a \in A. \tag{1}$$

Symmetric Games

A **symmetric game** is a game where the utility of individual agents is *invariant* under the permutation of agents. Formally, it is a game where $A_1 = A_2 = \ldots = A_N$, and for any permutation $\pi : \mathcal{I} \to \mathcal{I}$

$$u_i(a_1, \ldots, a_j, \ldots, a_N) = u_{\pi(i)}(a_{\pi(1)}, \ldots a_{\pi(j)}, \ldots a_{\pi(N)}). \tag{2}$$

Examples: Prisoner's Dilemma, selfish lightbulbs

Let $\varsigma_k = \{C \in \mathcal{I} \mid |C| = k\}$ be the set of coalitions of size k. The image of a permutation restricted to C is also a set of size k. Let $\pi_\sharp : \varsigma_k \to \varsigma_k$ be a map on coalitions induced my permutations such that $\pi_\sharp : (i_1, i_2, \ldots, i_k) \mapsto (\pi(i_1), \pi(i_2), \ldots, \pi(i_k))$. Since π is bijective, $\pi_\sharp$ is also bijective; therefore $\pi_\sharp$ is a permutation on coalitions.

We could construct a weighted network (V, E, W) representing **coalition relationships**. The coalitions are the nodes and an edge (i, j) is formed between coalitions i and j if they share at least one member. The weight function $W : V \times V \to \mathbb{N}_0$ computes the number of members shared by each coalition. We observe that the coalition relationships are invariant under permutations of agents.

We observe that the Co-Op coalition utility function carries forward the symmetry in u_i:

$$\begin{aligned}
u_{\pi_\sharp(c)}(a_\pi) &= \min_{i \in \pi_\sharp(C)} u_i(a_\pi) \\
&= \min_{i \in \pi_\sharp(C)} u_{\pi^{-1}(i)}(a) \\
&= \min_{j \in C} u_j(a) = u_c(a).
\end{aligned} \tag{3}$$

Fixed Points

A **fixed point** of an operator $\mathcal{K}$ is an element $\alpha \in A$ if $\mathcal{K}\alpha = \alpha$. Moreover it is an **attractive fixed point** if starting from an arbitrary element a_0 in some neighbourhood $B \subseteq A$ of α the sequence $\{a_t \mid a_{t+1} = \mathcal{K}a_t\}$ converges to α. N is called the attracting set of α.

In real life terms, if we can find $\mathcal{K}$ which has an attractive fixed point, we just need to loop *any* project proposal $a_0 \in B \subseteq A$ round the sequence over and over again and we are guaranteed to reach a stable agreement.

We could also find, for each a, the set of sequence operators that maps $a \mapsto a$: $\mathcal{O}_a = \{\mathcal{K} \in \mathcal{O} \mid \mathcal{K}a = a\}$, or the **stabiliser subgroup** or a. This quantifies the stability of each state against a coalition. It is worth remarking that it is quite possible that $\mathcal{O}_{a^*} \neq \Gamma$ i.e. for the optimum of Co-Op is *not* stable against coalition decision making; in the worst case, $\mathcal{O}_{a^*} = \{\mathcal{I}\}$.

There is actually a very simple but neat observation regarding coalition operators.

Theorem: Coalition operators map every element in state space to a fixed point, i.e. for a coalition operator $\mathcal{C}$, $a' = \mathcal{C}a$ is a fixed point of $\mathcal{C}$ for any $a \in A$, i.e. $\mathcal{C}a' = a'$. In other words, coalition operators are *idempotent*

$$\mathcal{C}^2 = \mathcal{C}.$$

Moreover, any state is either a fixed point or in the attracting set of a $\mathcal{C}$.

For a state space with S number of elements, there are $\sum_{k=0}^{S} \binom{S}{k} k^{S-k}$ number of idempotent functions.

Single agents

If each agent are acting as individuals, a state is a fixed point of any sequence of individuals if and only if it is the (pure) Nash equilibrium of the equivalent simultaneous game

$$u_i(a_i \mid a_{-i}) \geq u_i(a_i' \mid a_{-i}) \quad \forall\, a_i' \in A_i \quad \forall i \in \mathcal{I}. \tag{4}$$

Once you arrive at a fixed point (no matter how you arrive), the sequence of operations is irrelevant. A fixed point for any sequence is a fixed point for all sequences.

This is equivalent to the usual **sequential game** in game theory. We assume the game has **perfect information**, there is a **subgame perfect equilibrium** that can be calculated by backward induction.

Partition Set

A **partition set** partitions the set of agents into coalitions and assigns to each coalition a new utility function u_c, which is in some manner or another derived from the utilities of the constituent agents. Every agent is in one and only one coalition. We can think of the coalitions as coarse grained agents which interact in a sequential game.

Let the set of coalitions K be a cover of the set of agents. In the Co-Op philosophy, the utility of a coalition is

$$u_c(a_c \mid a_{-c}) = \min_{i \in C} u_i(a_c \mid a_{-c}). \tag{5}$$

A state is a fixed point of any sequence of a partition set of coalitions if and only if

$$u_c(a_c \mid a_{-c}) \geq u_i(a_c' \mid a_{-c}) \quad \forall\, a_c' \in A(C) \quad \forall C \in K. \tag{6}$$

In other words, the fixed point is a fixed point of all the individual coalition operators.

Overlapping Coalitions

Suppose $\mathcal{K}a = a$ is a fixed point of the coalition sequence $\mathcal{K}$. If a is a Nash equilibrium it will be a fixed point. However it may be the case that as a is cycled through the sequence again, it may change to other states but be returned to a. Moreover all points that a go through in the sequence are fixed points of sequences that are cyclic permutations of $\mathcal{K}$. So with every sequence of overlapping coalitions K we can associate with it a **limit cycle** of $|K|$ many elements. We may have more exotic p-cycles of coalitions s.t. $\mathcal{K}^n a = a$ for $n = p\mathbb{Z}$.

Other Fixed point formulations

We can flesh out this landscape of fixed points and attracting sets with two approaches. One is to construct a metric space (A, d), i.e. impose a measure of distance between states; the other is to represent our system of states and operators in terms of linear algebra.

Metric space approach

If we have a (non-empty, complete)[4] metric space (A, d), we can make use of the Banach fixed point theorem. If $\mathcal{K}$ is a contraction, i.e. $\exists q \in [0, 1)$ s.t. $\forall \, \alpha, \beta \in A, \; d(\mathcal{K}\alpha, \mathcal{K}\beta) \leq q d(\alpha, \beta)$, there exists a *unique* attractive fixed point $\mathcal{K}\alpha^* = \alpha^*$.

What remains is to choose a metric that sensibly models how There are two obvious choices for the metric $d : A \times A \to \mathbb{R}_+$. The first one is to embed $A \hookrightarrow \mathbb{R}^N$ (N being the number of agents) using the utility function as a coordinate map: $a_i = u_i(a)$. This induces a Euclidean metric on A. However, since the ideal situation is to solve the 'Co-Op' problem, we can make a legitimate simplification and choose $(a, a') \xrightarrow{d} |U(a) - U(a')|$, i.e. the difference between two states is their relative performance in optimising 'Co-Op'.

Linear Algebra Approach

We define $V(A)$, a real vector space taking elements of A as its basis $\{\varphi_a\}_{a \in A}$. Since $A = A_1 \times \cdots \times A_N$, our vector space is a tensor product space, allowing for a compact representation: $V(A_1 \times \cdots \times A_N) = V(A_1) \otimes \cdots \otimes V(A_N)$ where $\otimes$ is the *kronecker product*. Γ induces a set of *linear* operators L; if $a \xrightarrow{\mathcal{K}} a'$, there is a corresponding linear operator $l(\mathcal{K}) \in L$ such that $\varphi_a \xrightarrow{l(\mathcal{K})} \varphi_{a'}$. We have the commutative diagram:

$$
\begin{array}{ccc}
A & \xrightarrow{\;\mathcal{K}\;} & A \\
\downarrow & & \downarrow \\
V & \xrightarrow{\;l(\mathcal{K})\;} & V
\end{array}
$$

The action of the linear operator $l(\mathcal{K})$ can be represented by a matrix $\mathcal{K}$ (overloading notation since it is becoming ridiculously cumbersome?): entries $\mathcal{K}_{a'a} = 1$ if $\varphi_a \xrightarrow{l(\mathcal{K})} \varphi_{a'}$, and $\mathcal{K}_{a'a} = 0$ if otherwise. In other words, the a'^{th} row in the ath column is one if and only if the corresponding sequence operator sends state a to a', otherwise it is zero.

Matrices $\mathcal{K}$ are generated by matrices $\mathcal{C}$. Needless to say L is also a finitely represented semigroup closed under matrix multiplication. Since $\mathcal{C}^2 = \mathcal{C}$, coalition operators are *projectors* onto subspaces of V.

A fixed point of operator $\mathcal{K}$ is then simply the eigenvector of the matrix $\mathcal{K}$ with eigenvalue one. We can take a matrix and compute its singular/eigenvalues for a stability analysis of that operator.

    Comment: Given the particular structure of the matrices, analysis should be tractable. . .

Points to Explore:

- Decomposition of operators into tensor products: what is the significance? look at high order svd.

---

[4]Since A is finite we are fine.

Classification of Coalitions

Happy Coalitions

A **happy coalition** C with respect to a state a is one such that

$$u_i(\mathcal{C}a) \geq u_i(a) \quad \forall\, i \in C.$$

Suppose a coalition is happy for

> Good coalitions lead to good sequences vs Co-Op objective?

Summary Remarks

The really hard but interesting question is the relationship between the utilities of the agents u and the optimisation performance of sequences. Specifically, how *correlations* between decisions made by agents affect the *algebra* between coalition operators that generate the sequences. Some nice results should happen if agent decisions are uncorrelated/positively correlated/ anti-correlated. (We need some measure of correlation)

We expect computation and simulation to be expensive; complexity of computation mentioned so far is exponential in the number of states.

If we have time, We can hop onto the intellectual gravy train of machine learning and pretend some reinforced learning algorithm adaptively learning sequences is a good model of human behaviour. Further work can also be done on generalising the coalition operator e.g. do agents retain *memory* from previous coalition