LF-011

不確実な状況下における協調プラン探索法への通信の導入

Introducing Communication to Joint Policy Search Algorithm for Networked Distributed POMDPs

田崎 誠 † **籔 悠一** † **横尾 真** † **岩崎 敦** †
Makoto Tasaki Yuichi Yabu Makoto Yokoo Atsushi Iwasaki

1 はじめに

部分観測可能マルコフ決定過程 (Partially Observable Markov Decision Process, POMDP) は, 観測に不確実性が存在する場合，例えば移動ロボットが現在位置を正確に把握できない場合等に，望ましい行動計画を得るためのモデル化手法として用いられている [1]．POMDP における行動計画はポリシーと呼ばれ, エージェントが各信念状態で取るべき行動を示したマップで表される．

近年，不確実性が存在するマルチエージェントシステムをモデル化する手法として POMDP を用いる研究が盛んである．これまでに，エージェント間の相互作用の局所性をモデル化したネットワーク分散型 POMDP (Networked Distributed POMDP, ND-POMDP) [2]，および ND-POMDP において，適格性を満足する, つまり常に楽観的な値をとるヒューリスティック関数に基づく探索手法を利用することにより, 複数エージェントの最適なポリシーの組合せ (結合ポリシー) を求めるアルゴリズム (Search for Policies In Distributed EnviRonment, SPIDER) が提案されている [3].

エージェントがチームとして行動する場合でも，分散センサネットワークや分散無人機群等の応用においては [4], エージェント間の相互作用に強い局所性が存在する．例えば，非常に大規模な分散センサネットワークであっても，ある特定のターゲットを捕捉するために協力する必要のあるセンサは全体のごく一部である．ND-POMDP はこのようなエージェント間の相互作用の局所性を扱うことが可能であり，SPIDER を用いた場合, エージェント間の相互作用に十分な局所性が存在すれば，エージェント数が増加しても現実的な時間で結合ポリシーを得ることが可能となる．しかし，各エージェントのポリシーの大きさは，行動計画のステップ数 (ポリシーの長さ) に対して指数的となり，ステップ数が 4 より大きいポリシーを対象とすることは困難であった．

本研究では，この問題を解決するために，SPIDER にエージェント間通信を導入したアルゴリズム (SPIDER with Communication, SPIDER-Comm) を提案する．

従来手法では，各エージェントは整合の取れた初期信念状態を持って行動を開始するが，行動後に得る観測は異なるため，各エージェントの持つ信念状態は次第に異なるものとなる．SPIDER-Comm では, エージェントは定期的に通信を行い，観測履歴や行動履歴等の情報を共有する．この結果，各エージェントは整合の取れた単一の信念状態に到達し，その信念状態を初期状態とする新しいポリシーを構築することで，ポリシーの大きさの組合せ爆発を避けることが可能となる．

しかし, SPIDER への通信導入には以下の課題がある．

1. 通信を導入した場合，相互作用の局所性を利用することができず，従来の SPIDER のヒューリスティック関数が利用できない．
2. 通信後の信念状態の数が，ポリシーの長さに対して指数的となる．

課題の 1 を解決するため，本論文では，通信の結果を「利得が最も大きくなる信念状態を得る」と仮定することにより，適格性を満たすヒューリスティック関数を構築する方法を示す．

また，課題の 2 を解決するため，信念状態の取りうる空間を一定区間ごとに区切ったメッシュを用い, 各区間の中心点により近似するという手法を提案する．

2 ネットワーク分散型部分観測可能マルコフ決定過程

本論文で用いるネットワーク分散型部分観測可能マルコフ決定過程 (ND-POMDP) [2] のモデルを説明する．ND-POMDP は，POMDP において，エージェント間の相互作用の局所性を陽に表現したものであり，$\langle S, A, P, \Omega, O, R, b\rangle$ で定義される．

$S = \times_{1\le i\le n} S_i \times S_u$ は状態集合である．S_i はエージェント i の内部状態の集合であり，S_u はエージェントの行動に影響を受けない状態の集合，例えば天候やセンサネットワークにおけるターゲットの位置等を示す．

$A = \times_{1\le i\le n} A_i$ はエージェントの行動の組合せの集合である．A_i はエージェント i のとりうる行動の集合を表す．

$P(\vec{s}, \vec{a}, \vec{s'}) = P_u(s_u, s'_u) \cdot \prod_{1\le i\le n} P_i(s_i, s_u, a_i, s'_i)$ は状態の遷移関数であり，状態 $\vec{s} = \langle s_1, \ldots, s_n, s_u\rangle$ において $\vec{a} = \langle a_1, \ldots, a_n\rangle$ という行動をとった場合に，状態 $\vec{s'} = \langle s'_1, \ldots, s'_n, s'_u\rangle$ に遷移する確率を返す．

$\Omega = \times_{1\le i\le n} \Omega_i$ はエージェントの得る観測の組合せの集合である．Ω_i はエージェント i が得る観測の集合を表す．

$O(\vec{s}, \vec{a}, \vec{\omega}) = \prod_{1\le i\le n} O_i(s_i, s_u, a_i, \omega_i)$ は観測関数であり，状態 $\vec{s}$ において行動 $\vec{a}$ をとったとき，観測 $\vec{\omega} = \langle \omega_1, \ldots, \omega_n\rangle \in \Omega$ を得る確率を返す．

$R(\vec{s}, \vec{a}) = \sum_l R_l(s_{l_1}, \ldots, s_{l_r}, s_u, \langle a_{l_1}, \ldots, a_{l_r}\rangle)$ は報酬関数である．l は相互作用のあるエージェントの部分集合を表しており，$r = |l|$ である．報酬関数に基づいて，グラフ $G = (Ag, E)$ が定義される．Ag はノードであり，各エージェントを表す．E はエッジであり，$E = \{l | l \subseteq Ag \wedge R_l\}$ と定義される．

b は初期信念状態を表し，$b(\vec{s}) = b_u(s_u) \cdot \prod_{1\le i\le n} b_i(s_i)$ と定義される．b_u は s_u の確率分布であり，b_i はエージェント i の内部状態 s_i の確率分布を表す．

† 九州大学大学院システム情報科学府, Graduate School of ISEE, Kyushu University

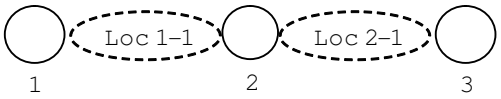

図 1　センサネットワーク

本研究の目標は, 初期信念状態 b, 最大ステップ数 T に対してシステム全体の報酬を最大化する結合ポリシー $\pi = \langle \pi_1, \ldots, \pi_n \rangle$ を探索することである.

分散センサネットワークを ND-POMDP としてモデル化した例を示す [2]. 3 つのセンサ (エージェント) が存在する場合の例を図 1 に示す. ある時刻において，それぞれのセンサは 4 方向 (東西南北) の内の 1 方向をサーチすることができる. その際, センサはサーチした場所に探索すべきターゲットが存在するか否かの観測を得る. ターゲットは 2 体存在し, それぞれ独立して行動する. 図 1 の Loc1-1 と Loc2-1 はターゲットが現れうる場所を示す. ターゲット 1 (2) は Loc1-1 (Loc2-1) に現れるかもしくはどこにも現れない. 2 つのセンサが同時にターゲットをサーチした時, 発見したとみなし, 報酬が得られる. それぞれのセンサが得る観測やターゲットの行動は他のセンサの行動 (サーチの方向) には依存しない.

SPIDER は ND-POMDP において最適な結合プランを探索するアルゴリズムである．このアルゴリズムでは, エージェント間の相互作用を表すネットワークから深さ優先探索木 (Depth-first Search Tree, DFS 木) を構築し, この DFS 木を利用してヒューリスティック探索を行う. それぞれのエージェントが持つポリシーに関して, 他のエージェントは観測に不確実性が存在しない (マルコフ決定過程である) と仮定して，適格性を満たす推定値を求め, この推定値を用いて探索木の枝刈りを行う.

3　エージェント間通信を用いたプラン構築アルゴリズムの提案

3.1　基本的なアイデア

SPIDER では，各エージェントの持つポリシーは，分岐していく木として表現される. 初期信念状態を起点として, 観測の種類の数だけ枝分かれし, それぞれの観測に対して選択すべき行動がマッピングされている. よって, ポリシー木の大きさはポリシーの長さ (行動の回数) に対して指数的となる.

SPIDER-Comm では，エージェントは以下のように通信を行う.

- エージェントは, あらかじめ定められた一定の間隔で通信を行う．すなわち, 通信を行うフェイズと, 通信を行わず通常の行動を行うフェイズに分かれる.
- 通信を行うフェイズでは，全エージェント間で行動／観測履歴の交換が行われ，整合の取れた新しい信念状態が得られる.

通信後の各信念状態において，新しい結合ポリシーを定義することにより，ポリシーの大きさによる組合せ爆発を回避することが可能となる.

しかしながら，通信を導入した場合，ネットワーク構造の局所性を利用することができず，SPIDER のヒューリスティック関数が利用できないという問題が生じる. 図 1 の例を用いて説明すると, センサ 1 とセンサ 3 には直接の相互作用はない. SPIDER では，まずセンサ 2 のポリシーを固定し，そのポリシーに対して最適反応となるセンサ 1 およびセンサ 3 のポリシーを求めるという順序で探索が行われる. この際, センサ 2 のポリシーが固定されていることから, センサ 1 のポリシーとセンサ 3 のポリシーは独立に計算できる．しかしながら, エージェント間の通信を考慮すると, センサ 1 のポリシーを評価する際には, 通信後の信念状態がどうなるかを考慮する必要があり, そのためにはセンサ 3 のポリシーを考慮する必要が生じる.

本論文では, 通信後の状態に関して, センサ 3 のポリシーによって到達する信念状態が, 通信後の利得を最も大きくする場合を仮定することにより，適格性を満足するヒューリスティック関数を構築する．この方法により，センサ 3 のポリシーが未決定な時点において，センサ 1 のポリシー選択に関して適格性を満足するヒューリスティック関数の構築が可能となる.

また，通信結果により得られる各信念状態において, 新しい結合ポリシーを定義することにより，各ポリシーのサイズ自体は小さくなるが，結合ポリシーを準備しておく必要のある信念状態の数は，全体の行動計画の長さに対して指数的に増加する．そこで, SPIDER-Comm では信念状態が取りうる空間に対して, 一定の範囲ごとに区切ったメッシュを用い, 各範囲にある信念状態を一つの点に近似することで考えるべき信念状態の数を減らすというアイデアを用いる.

3.2　アルゴリズムの詳細

本節では, エージェント間通信を用いたプラン構築アルゴリズム (SPIDER with Communication, SPIDER-Comm) の詳細を示す．ここで便宜上, ツリー及びポリシー, 期待利得に関して以下の表記法を使用する. i, j はエージェントの ID を表す：

π^{i+} ⇒　i をルートとするサブツリー全体の結合ポリシー.
π^{i-} ⇒　i の上位ノードに関する結合ポリシー.
π_i ⇒　i のもつポリシー.
$\hat{v}[\pi_i, \pi^{i-}]$ ⇒　π_i 及び π^{i-} に対する π^{i+} での期待利得の上限.
$\hat{v}_j[\pi_i, \pi^{i-}]$ ⇒　i の子ノード j による π^{i+} での期待利得の上限.
$v[\pi_i, \pi^{i-}]$ ⇒　π^{i-} に対する π_i による期待利得.
$v[\pi^{i+}, \pi^{i-}]$ ⇒　π^{i-} に対する π^{i+} による期待利得.
$\ddot{v}[b, n]$ ⇒　n 回目の通信によって, 全エージェントの信念状態が b となったときのそれ以降の期待利得.

SPIDER-Comm が与えるポリシーは, 通信フェイズと行動フェイズに対応して構成される. 行動フェイズのポリシーは, k ステップの行動のマップで形成される. 通信フェイズのポリシーは, 状態に関わらず 1 ステップの通信で形成される. 各エージェントは通信を行い観測履歴と行動履歴を交換し, この結果得られる信念状態を新たな初期状態とする. 次に k ステップの非通信フェイズがあり, 行動フェイズの後通信フェイズによって通信が行われる. このように, 非通信フェイズと通信フェイズを交互に繰り返す. 非通信フェイズを k ステップ, 通信フェイズの回数を m 回とすると最大ステップ数 T は

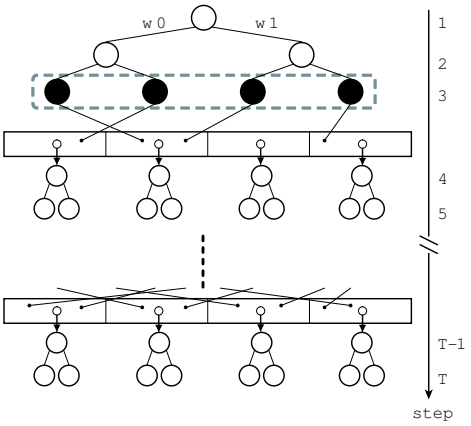

図 2　SPIDER-Comm の計算方針

$T = (k \times (m+1)) + m$ となる.

次に SPIDER-Comm の簡単な流れを示す. 図 2 は, 通信を行う場合の各エージェントが持つポリシー木の例である. ここで破線からなる四角は通信フェイズを表し, その下の四角は, 信念状態に関する確率の範囲区分を表す. 黒円から延びる線は, 通信結果から得られた状態に関する確率分布である. SPIDER-Comm ではまずポリシー全体を通信フェイズごとに分け, 最下部からの探索が行われる. 各範囲ごとに定められた確率分布から, SPIDER の手法で最下部 (図 2 ではステップ $T-1, T$ に関して) の最適なポリシーが決定される. つまり, 各範囲に到達した時の期待利得が決定された事になる.

次に, 1 つ上の範囲についてポリシーの決定を行う. ここで得られる期待利得は, 通常のポリシーから計算される期待利得と通信後の到達範囲に関して先ほど求めた期待利得を足したものである. よって SPIDER の手法による探索の際に, ヒューリスティック値に通信以降に見込める最も高い期待利得を足し合わせたものを用いる. 更に 1 つ上の範囲に関して, というように徐々にポリシー木を遡って探索し, 最終的なポリシーを決定する.

SPIDER-Comm のアルゴリズムを Algorithm 1 に示す. 引数にある *CommPhase* は何回目の通信フェイズについて考えているかを表している. SPIDER-Comm は動的計画法を用いており, 最後の k ステップから計算する. まず, makeStarting 関数によって信念状態の近似点の集合を作る (3 行目). そして, $b \in B$ となる全ての信念状態 b を初期状態とする最適ポリシーを求め, $(b, CommPhase)$ のペアに対する k ステップの結合ポリシーを $\pi^*[b, CommPhase]$ に格納する. これを, *CommPhase* が 0 になるまで繰り返す.

次に, 実際にポリシーを求める Algorithm 2 について説明する. Algorithm 2 は, SPIDER の手法を取り入れ, ヒューリスティック関数を用いて初期状態が b となるときの, 大域的最適ポリシーを求める. まず, Π_i に現在の k ステップにおける取りうるすべてのポリシーを格納する (2 行目). エージェント i が葉であるとき, π^{i-} はすでに決まっているので, GETVALUE 関数によって利得を計算する (5 行目). エージェント i が葉でないとき, まず $\hat{\Pi}$ にヒューリスティック値によってソートした, ポリシーの集合を代入する (11 行目). UPPER-BOUND-SORT アルゴリズムでは, ヒューリスティック値の計算と, それを元にした全てのポリシーに関する降順のソートが行われる. エージェント i のポリシーを決め, 葉エージェントの最適なポリシーを探索する (16-20 行目). 最後に最大となる期待利得とそのときのポリシーを保存する (21-23 行目).

3.3　提案アルゴリズムの最適性

通信後の信念状態の近似を行わない場合，SPIDER-Comm のヒューリスティック関数は適格，すなわち常に楽観的な値となっており，実際に得られる期待利得を上回ることはないことが証明できる．よって，SPIDER-Comm 内の枝刈りによって最適性が失われることはなく，SPIDER-Comm は必ず最適解を得ることが示される．一方，信念状態の近似を行う場合には，中心点での最適プランが区間全体での最適プランとならない可能性があり，最適性は保証されない.

Algorithm 1 SPIDER-Comm($i, CommPhase$)

```
1: initialize v̈, π* ← null, val ← 0
2: while CommPhase ≥ 0 do
3:    B ←makeStarting(CommPhase)
4:    for all b ∈ B do
5:       π*[b, CommPhase], val ←
         FINDPOLICY(b, root, null, −∞, CommPhase, v̈)
6:       v̈[b, CommPhase] ← val
7:    CommPhase − −
8: return π*
```

Algorithm 2 FINDPOLICY($b, i, \pi^{i-1}, threshold, Comm, \ddot{v}$)

```
1: π^{i+,*} ← null
2: Π_i ← GET-ALL-POLICIES(k, A_i, Ω_i)
3: if IS-LEAF(i) then
4:    for all π_i ∈ Π_i do
5:       v[π_i, π^{i−}] ← GETVALUE(b, π_i, π^{i−}, v̈)
6:       if v[π_i, π^{i−}] > threshold then
7:          π^{i+,*} ← π_i
8:          threshold ← v[π_i, π^{i−}]
9: else
10:   children ← CHILDREN(i)
11:   Π̂_i
      ← UPPER-BOUND-SORT(b, i, Π_i, π^{i−}, Comm, v̈)
12:   for all π_i ∈ Π̂_i do
13:      π̃^{i+} ← π_i
14:      if v̂[π_i, π^{i−}] < threshold then
15:         Go to line 12
16:      for all j ∈ children do
17:         jThres ← threshold − v[π_i, π^{i−}]
            − Σ_{k∈children,k≠j} v̂_k[π_i, π^{i−}]
18:         π^{j+,*} ← FINDPOLICY(b, j, π_i||π^{i−}, jThres, Comm, v̈)
19:         π̃^{i+} ← π̃^{i+}||π^{j+,*}
20:         v̂_j[π_i, π^{i−}] ← v[π^{j+,*}, π_i||π^{i−}]
21:         if v[π̃^{i+}, π^{i−}] > threshold then
22:            threshold ← v[π̃^{i+}, π^{i−}]
23:            π^{i+,*} ← π̃^{i+}
24: return π^{i+,*}, threshold
```

3.4 提案アルゴリズムの計算量評価

最大ステップ数 T, エージェント数 n, 行動の数 $|A|$, 観測の数 $|\Omega|$ である場合について考える. 通信を行わない場合, ポリシーはステップ数毎に観測の数だけ分岐するツリーにより表記される. 従って, ポリシーのサイズは $|\Omega|^T$ である. これに対し, k ステップごとの通信を行う場合は, 近似点の数を m, 通信回数を c $(= T/k - 1)$ と置くと, ポリシーは通信フェイズ毎に, 観測の数に等しい分岐が k 回あるツリーを作り, ポリシーの合計サイズは $(cm + 1) \cdot |\Omega|^k$ となる. 以上より, ポリシーのサイズは, 通信を行わない場合では最大ステップ数 T に関して指数オーダであるのに対して，通信および通信後の状態の近似を行う場合では，m を定数とすれば多項式オーダとなることが示される.

また, 最悪計算量は, 通信を行わない場合では最大ステップ数 T に関して $(|A|^{|\Omega|^T})^n$ であり 2 重指数オーダとなる. これに対し, 通信を行う場合の最悪計算量では T に関して $(cm + 1)(|A|^{|\Omega|^k})^n$ であり多項式オーダとなる.

4 計算機実験による評価

本実験では図 1 で示す 3 エージェントのセンサネットワークを用いる．この例題では，s_u の取りうる値は，ターゲットが両方に存在する，どちらか片方にのみ存在する，どちらにも存在しない場合の 4 通りである．また，ターゲット 1, 2 それぞれの発見時の利得を +45, +35 とし, 未発見時の利得を −5 とする.

まず, 表 1 に，SPIDER-Comm において，メッシュの分割数が計算時間及び期待利得に与える影響を示す. 最大ステップ数 $T = 8$, $k = 2$ (通信回数は 2 回) とした場合に, メッシュの分割数を変化させた場合の期待利得と計算時間を示す．分割数を増すごとに期待利得は大きくなるが, それに伴う計算時間の上昇も著しいことが示されている．以下では各状態毎に 10 区分として評価を行う．次に, 図 3 に $k = 2$ の場合の計算時間の比較を表す. SPIDER と SPIDER-Comm では求めているプランが異なるため，単純な実行時間の比較はできない. そこで, 比較の対象として，大域最適解探索アルゴリズム (Global Optimal Algorithm, GOA) [2] を拡張した GOA-Comm (GOA with communication) を用いる. このアルゴリズムは, 通信により到達可能な各信念状態から, 全探索により最適なポリシーを探索するアルゴリズムである.

表 1　メッシュの分割数と計算時間及び期待利得の比較

各状態の分割数	計算時間 [msec]	期待利得
2	1703.00	312.58
10	127890	319.67
20	846750	321.03

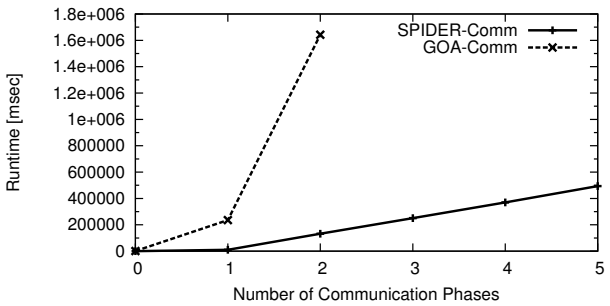

図 3　提案アルゴリズムと全探索アルゴリズムの計算時間の比較

図 3 のグラフの横軸は通信の回数を, 縦軸は計算時間を表す. 通信 n 回の時, $k = 2$ であるため最大ステップ数は $T = 3n + 2$ となる．SPIDER-Comm ではヒューリスティック探索の導入により，GOA-Comm と比較して大幅な計算時間の削減が得られることが示されている.

SPIDER では，現実的な時間内で得られるポリシーの長さは 4 が限界であるのに対して [3]，これらの評価結果より，SPIDER-Comm では，通信の間隔が十分小さく，メッシュの数が十分小さければ，ポリシーの長さ自体は，数十ステップまで大きくすることが可能であることが示されている.

5 結論

本論文では，不確実な状況下での協調プラン探索法における，エージェント間通信の導入法について検討した．具体的には，従来手法である SPIDER で生じるポリシーの大きさの組合せ爆発という問題を避けるため，エージェントが定期的に通信し，新しい信念状態からポリシーを実行するという方法 (SPIDER-Comm) を提案した．本手法は，通信の導入によって生じる，新しいヒューリスティック関数の構成，および通信後の信念状態数の組合せ爆発という二つの課題を解決している．従来の SPIDER では現実的な時間内において計算不可能であった長さのポリシーが，本アルゴリズムの実装により計算可能となることを示した.

謝辞

本研究を進めるにあたり，日本学術振興会科学研究費補助金基盤研究 (B)（課題番号 17300049）の助成を受けました．ここに深く感謝いたします.

参考文献

[1] W. Lovejoy. Computationally feasible bounds for partially observed markov decision processes. *Operations Research*, Vol. 39, pp. 162–175, 1991.

[2] R. Nair, P. Varakantham, M. Tambe, and M. Yokoo. Networked distributed POMDPs: A synthesis of distributed constraint optimization and POMDPs. In *Proc. of 12th National Conf. on Artificial Intelligence (AAAI-05)*, 2005.

[3] P. Varakantham, J. Marecki, Y. Yabu, M. Tambe, and M. Yokoo. Letting loose a SPIDER on a network of POMDPs: Generating quality guaranteed policies. In *Proc. of 6th Int. joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS-07)*, 2007.

[4] D. V. Pynadath and M. Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of Artificial Intelligence Research*, Vol. 16, pp. 389–423, 2002.