

# マルチエージェント巡回清掃における直接的な通信を用いた 分業促進手法

Method of Promoting Division of Labor by Using Communication for Multi-agent Continuous Cleaning

杉山 歩未\*<sup>1</sup>  
Ayumi Sugiyama

Sea Vourchteang\*<sup>1</sup>

菅原 俊治\*<sup>1</sup>  
Toshiharu Sugawara

早稲田大学 基幹理工学研究科 情報理工・情報通信専攻  
Department of Computer Science and Communications Engineering, Waseda University

In this paper, we proposed the method to promote division of labor by using simple communication between agents. We already showed that sharing information are not always best solution. So, we tried to find simple communication that improve efficiency of system. We paid attention to amount of task, center node of task, and search range. And then, we could improve the efficiency of system and show that our proposed method promoted division of labor. Interestingly, with our proposed method, agents seem to make a team without algorithm for team organization.

## 1. 序論

複数のロボットや計算機が自律的に協調して作業や処理を行うことが期待されている。例えば広い空間での清掃や警備、惑星探索などへのロボットの活用があげられる。このような時間制限のある作業や広大な領域での作業は、能力の制限から複数のロボットによる共同作業が必要となる。しかし、共同作業では情報交換、衝突回避、分業、地図生成、他者の振る舞いを考慮した行動戦略の決定など多くの要因を考慮する必要があり、効率的な作業のための協調メカニズムについての最適解はいまだ自明でない。また、多様な情報を活用するには情報共有のコストとのトレードオフが生じる。本研究では、独立な行動主体をエージェントにとらえ、巡回清掃問題を題材に、単純化した内部表現や通信を用いながらもなるべく効率のよい協調作業ができるアプローチに着目して、自律的行動決定と通信による協調手法の提案を行った。

## 2. 関連研究

本稿で想定するような複数エージェントによる清掃やパトロールに関する研究は多く存在する。複数者の共同作業において、適切な分業を行うことは重要な要素の1つである。このような複数者による分業のための協調戦略は大別して2つの方法がある。1つは領域を分割し、各エージェントに担当領域を割り当てるアプローチである。もう1つは領域を分割せず、各エージェントが協調のための巡回戦略を学習、選択するものである。前者の手法として、[1]では複数ロボットがイベントの発生確率とその場所を訪問可能な頻度の情報を交換し、担当領域を分割する手法を提案している。また、[2]では各担当領域の広さを風船の圧力としてモデル化し、圧力が均等になるように担当領域の広さを交渉する手法が提案されている。このような領域分割による分業は、競合の発生を防ぐには非常に効率的であるが、エージェント数が多い場合は領域の交渉に時間がかかることや、複数台での同時作業が求められるタスクが発生した場合にすばやく対応できないといった課題がある。もう1つの、領域分割を行わず協調戦略を学習する手法として、[3]では重力のアイデアを用い、長時間訪問していない場所は強い

重力をもつとして、エージェントがそういった場所を優先的に訪れるアルゴリズムを提案している。他にも、[5]ではいくつかの簡単な戦略から適切な戦略を学習する手法を提案し、環境の特徴によってエージェントは異なった戦略のバランスに分かれることで清掃効率を向上させることを示した。我々も[4]において[5]を拡張し、環境内のイベントの発生確率の学習手法を提案し、同時作業では事前に情報を与えない方が自然な分業を発生させ、清掃効率を向上させる場合があることを示した。

[4]では情報共有のコスト極力減らすため、エージェント間通信を行わなかった。しかし、簡易な情報交換が分業を大きく促進させることが現実の世界でも見受けられる。そこで本研究では[4]を拡張し、ロボットが簡単な通信を行える環境において人のように少ない情報の交換で効率の良い分業体制を促進する手法の提案を目的とする。

## 3. モデル

### 3.1 環境

本研究は[4]を拡張したものであるため、詳細についてはこちらも参考にされたい。清掃を行う環境をグラフ  $G = (V, E)$  で表し、ノードの集合を  $V = \{v_1, \dots, v_m\}$  とし、 $v$  は座標  $(x, y)$  をもつ。このノード上にエージェントやごみが存在する。 $E$  はノードをつなぐエッジ  $e$  の集合であり、エージェントはエッジによってつながれた隣接ノードに移動できる。便宜的にごみの溜まらないダミーノードを追加して、エッジの長さは全て1と単純化できる。ノード  $v_k$  から  $v_l$  への最短距離を  $d(v_k, v_l)$  と表わす。また、最小単位を1ステップとする離散時間を導入する。環境内の各ノードの汚れやすさを確率的に表現し、各ノード  $v$  のごみの発生確率の集合を  $\{P_v | v \in V, 0 \leq P_v \leq 1\}$  とおく。各ノードの汚れやすさを表す確率を以後塵の蓄積確率 (Probability of Dust Accumulation, PDA) と呼ぶ。各ノードのPDAの差が汚れやすさに偏りのある環境を表現する。

本研究では、[4]を拡張し、エージェント間の通信を導入する。ただし、[4]において述べられているように、単純な情報共有が作業効率の向上につながらない場合もあることを注意されたい。エージェント  $i$  からの距離が  $d_{com}$  内のノードを通信可能範囲  $V_{range}^i$  として以下のように定義する

$$V_{range}^i = \{v | d(v_i^i, v) < d_{com}\}. \quad (1)$$

連絡先: 杉山 歩未, 早稲田大学 基幹理工学研究科 情報理工・情報通信専攻, sugi.ayumi@ruri.waseda.jp

また、エージェント  $i$  は任意のエージェント  $j$  と最後に通信を行った時刻  $T_{last}^{i,j}$  を記録する。そして、情報の更新が少ない短時間での通信を避けるため通信間隔  $T_{limit}^i$  をもち、最後に通信してから現在までの時刻の間隔  $I_{com}^{i,j} = t - T_{last}^{i,j}$  が  $T_{limit}^i$  以上のとき、通信を行うこととする。本実験では  $d_{com} = 5$ ,  $I_{com} = 10800$  とした。本研究では簡易化のため通信時の情報損失は 0 とし完全な情報が通信できるとした。また、エージェントの位置情報は近年の測定技術の発展により比較的正確かつ容易に取得できるようになっているため、本研究では他のエージェントの位置は赤外線素子や環境インフラからのブロードキャストの利用を想定し観測可能とした。

### 3.2 エージェント

$A = \{1, \dots, n\}$  をエージェントの集合とし、時刻  $t$  における  $i$  の位置を  $v_t^i$  と表す。エージェント  $i$  は、PDA の値  $P_v$  を事前に知らず、それぞれが独立にその値を保持、学習する。その値を  $P_v^i$  とし、その初期値は 0 とする。また、エージェント  $i$  は有限容量のバッテリーをもち、充電が切れる前に充電基地に戻り充電を行う。

時刻  $t$  における各ノード  $v$  のごみの量を  $L_t(v)$  と表す。ごみの量は各ステップごとに  $P_v$  の確率で更新する。ごみは 1 ずつ発生し、回収されるまで蓄積されていく。時刻  $t$  においてあるエージェントが  $v$  を訪れた時、ごみは回収され  $L_t(v) = 0$  となる。エージェント  $i$  は各ノードが最後に清掃された時刻  $t_{visit}^v$  から、時刻  $t$  における各ノードのごみの存在期待値を  $EL_t^i(v)$  を、それまで学習した  $P_v^i$  の値を用いて

$$EL_t^i(v) = P_v^i(t - t_{visit}^v) \quad (2)$$

と計算する。また、清掃時に回収したごみ量と最後に清掃されてからの時刻の間隔  $I_t^i(v)$

$$I_t^i(v) = t - t_{visit}^v, \quad (3)$$

を用いて、以下のようにそのノードの  $P_v^i$  を学習する。

$$P_v^i \leftarrow (1 - \alpha)P_v^i + \alpha \frac{L_t(v)}{I_t^i(v)}. \quad (4)$$

ここで  $\alpha$  ( $0 < \alpha \leq 1$ ) は学習率であり、本実験では、 $\alpha = 0.05$  とした。

### 3.3 エージェントの探索戦略

以下エージェントが目標を決定するための戦略について簡単に説明するが、詳細については [4] を参照されたい。

#### 3.3.1 目標選択戦略

(1) ランダム法：全てのノードからランダムに目標ノードを決定する。(2) 貪欲法：各ノードのごみの推定値  $EL_t^i(v)$  の上位  $N_g$  個からランダムに 1 つの目標頂点を決定する。(3) 訪問間隔優先法：全てのノードのなかから、最後に清掃された時刻が古い上位  $N_i$  個からランダムに 1 つの目標頂点を決定する。[4] における斥力法と異なり、他者の位置情報が不明となった場合でも自然に拡張できる。(4) 戦略型目標決定法：近隣にごみが多いと判断したときは近隣を優先的に巡回する戦略である。

#### 3.3.2 学習型目標決定法 [5]

第 3.3.1 節の目標選択戦略から適切と思われる戦略  $s$  を学習する手法である。戦略  $s$  にしたがって目標のノードまで移動しながら清掃した際の、1 ステップあたりのごみの回収量を報酬  $u$  とし、各戦略  $s$  の行動価値を以下の式で学習する

$$Q^i(s) \leftarrow (1 - \alpha)Q^i(s) + \alpha \cdot u_{t_0, t_0 + d_{travel}}. \quad (5)$$

$d_{travel}$  は  $v_{target}^i$  に至るまでの所要時間、 $\alpha$  ( $0 < \alpha \leq 1$ ) は学習率である。エージェントはこの行動価値  $Q^i(s)$   $\epsilon$ -greedy 法によって目標選択戦略を決定する。

## 4. 提案手法

本手法は簡単な通信によって分業を促進させる手法を提案し、低コストながら効率の良い協調の実現を目的としたものである。以後本提案手法を AMTDS/LD を拡張させた *AMTDS/LD and Promoting division of labor* (AMTDS/LDPD) と呼ぶ。はじめに、エージェント  $i$  が充電基地に戻った際に、自分が学習した PDA の上位  $N_{search}^i$  個のノードを探索ノードの集合  $V_{search}^i$  として定義する。エージェントは 3.3.1 節における貪欲型および戦略型目標決定法によって目標頂点を選択する際  $V_{search}$  から選択を行う。また、この時に自身の探索ノード  $V_{search}^i$  に含まれるノードの PDA 値の和をエージェント  $i$  が責任をもつ仕事量  $PDA_{sum}^i$  として以下の式で計算する

$$PDA_{sum}^i = \sum_{v \in V_{search}^i} P_v^i \quad (6)$$

さらに、探索ノードにおける重心ノード  $v_{center}^i = (v_{centerX}^i, v_{centerY}^i)$  を以下のように決定する

$$v_{centerX}^i = \left[ \sum_{v \in V_{search}^i} \frac{P_v^i}{PDA_{sum}^i} v_x \right] \quad (7)$$

$$v_{centerY}^i = \left[ \sum_{v \in V_{search}^i} \frac{P_v^i}{PDA_{sum}^i} v_y \right]. \quad (8)$$

ここで、 $v_x, v_y, v_{centerX}^i, v_{centerY}^i$  は整数で表される座標である。

#### 4.1 仕事の分配

通信時に重心ノード  $v_{center}$ 、探索ノードの重さ  $PDA_{sum}$  の情報を交換する。その後、探索マップの重さを比較し、

$$1 + C_{th} < \frac{PDA_{sum}^i}{PDA_{sum}^j} \quad (9)$$

だった場合、エージェント  $j$  が責任をもつ仕事量は少ないとして、エージェント  $i$  は  $V_{search}^i$  のうち、 $P_v^i$  の低いノードから順に  $N_{give}$  個のノードに対し、以下のように  $P_v^i$  の情報をエージェント  $j$  に渡す

$$P_v^j \leftarrow P_v^j + \frac{1}{2} P_v^i \quad (10)$$

$$P_v^i \leftarrow \frac{1}{2} P_v^i. \quad (11)$$

ここで、 $C_{th}$  は閾値であり、今回は  $C_{th} = 0.05$  とした。ただし、情報を渡すノード  $v$  は  $d(v_{center}^i, v) > d(v_{center}^j, v)$  となる、相手のほうが重心に近いノードに限る。また、 $N_{give}$  は責任をもつ仕事量  $PDA_{sum}$  の差に応じて、以下の式で決定する

$$N_{give} = \max(N_{giveMax}, PDA_{sum}^i / PDA_{sum}^j \times 10). \quad (12)$$

ここで上限  $N_{giveMax}$  を設けるのは情報の与えすぎによる大きな変動を避けるためである。

情報交換が終わった後、情報を与えたエージェント  $i$  と受け取った  $j$  は探索範囲の広さを表す  $N_{search}$  を以下の式で更新する

$$N_{search}^i \leftarrow N_{search}^i - N_{give} \quad (13)$$

$$N_{search}^j \leftarrow N_{search}^j + N_{give}. \quad (14)$$

これは、仕事量が多いと思われるエージェントは他のエージェントに仕事を分配し、探索範囲を狭めることで特定の範囲に集中させ、逆に仕事量の少ないエージェントは情報を受け取った後、探索範囲を広げることで受け取った情報も含め広い範囲を探索させることを表わしている。これにより、一部のエージェントは狭い範囲を集中して訪問することにより、その範囲の正確なごみの発生確率を学習することでより精度の高い巡回が行えるようになり、別のエージェントは広範囲を探索することで環境全体を取りこぼしのないように巡回できる。

## 4.2 担当ノードの改善

4.1節はエージェント間に大きな仕事量の差があった場合の処理である。しかし、学習と4.1節の情報交換により各エージェントの仕事量と探索範囲が安定した後に、さらに分業を促進させるための別の処理を導入する。自分の他にあるノードの近辺を頻繁に訪れるエージェント  $j$  が居た場合、 $j$  の方がそのノードに訪れるコストが小さいと言える。これは、経済学の分野で用いられる機会損失という考え方に似ている。自分が責任をもっているが遠い場所を訪れるために、重心により近く精度の高い巡回ができる場所に集中する機会が失われることとなるため、このような機会損失の大きなノードは相対的に損失の小さい他のエージェントに任せることが望ましい。そこで、通信時にエージェントが仕事量の比較を行った際に以下の条件を満たすかを判定する

$$1 - C_{th} < \frac{PDA_{sum}^i}{PDA_{sum}^j} < 1 + C_{th}. \quad (15)$$

この条件を満たす時、両エージェントの仕事量は同等だと判断する。その後、探索ノード  $V_{search}$  のうち  $P_v$  が大きいノードから順に  $d(v_{center}^i, v) > d(v_{center}^j, v)$  となるノード  $v$  の情報を最大  $C_{sup}$  個まで式 10 と同様に渡し、その後そのとき渡したノード数を  $C_{give}$  として式 13 と同様に以下のように探索範囲を増減する

$$N_{search}^i \leftarrow N_{search}^i - C_{give} \quad (16)$$

$$N_{search}^j \leftarrow N_{search}^j + C_{give}. \quad (17)$$

本手法では探索範囲に差が生まれる仕組みを導入することで、同様の仕事量を担当していてもある部分に特化する者と、広範囲を幅広く探索する者を生み出すことを狙いとしている。4.1節のように仕事量の大きさに差があるときは下位のノードから交換し、同等のときは上位のノードから交換する理由は、学習が安定していない前半は仕事量を均等にするという単純な分業を狙いとするのに対し、安定後は大きな機会損失のあるノードを考慮して、分業の精度を向上させることを目的としているためである。

## 5. 実験

本節では評価実験の結果を示す。実験では、提案手法と2つの既存手法との比較を行った。1つめの既存手法は [5] で提案された全エージェントが全てのノードの  $P_v$  を事前に知っている AMTDS 手法であり、2つめの既存手法は [4] で提案したエージェントが  $P_v$  の値を事前に知らず、学習によって各エージェントが独立にその値を保持する AMTDS/LD 手法である。ただし、[4] とあらかじめ与える戦略の一部が変更されているため、結果にも多少の違いが現れていることに注意されたい。

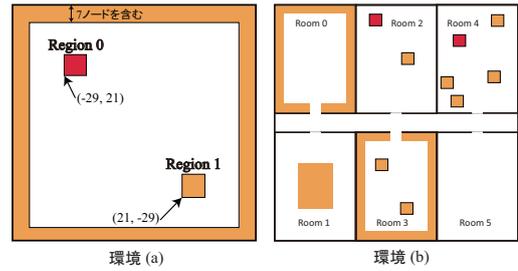


図 1: 環境

表 1: 各目標選択戦略のパラメータ

目標選択戦略	パラメータ	値
貪欲法	$N_g$	5
訪問間隔優先法	$N_i$	5
学習型目標決定法	$\alpha$	0.1
	$\epsilon$	0.05
AMTDS/LDPD	$N_{givemax}$	100
	$C_{sup}$	10

### 5.1 実験環境

エージェントが清掃する空間を  $101 \times 101$  の 2 次元グリッドとする。ノードの座標である  $(x, y)$  は  $-50 \leq x, y \leq 50$  の範囲と定める。全エージェントの充電基地は  $v_{base} = (0, 0)$  とし、エージェントは充電基地からスタートし、バッテリーが切れる前に基地に戻り充電を開始する。

実験を行う環境として図 1 に示す汚れやすさに偏りのある 2 つの環境 (a), (b) を用意した。環境のノードの PDA は図と対応して、 $P_v = 10^{-3}$  (赤),  $10^{-4}$  (オレンジ),  $10^{-6}$  (その他) とした。環境 (a) は壁の周囲と一部領域 (図中では Region とし示す) が汚れやすい環境。環境 (b) は、図中で Room と示す部屋のように環境が分断されている複雑な環境である。

本実験では 1 回の試行は 3000000 ステップとし、1 ステップでエージェントは隣接ノードへの移動、ごみの回収を行う。結果は 20 試行の平均値を用いた。環境には 20 台のエージェントを導入し、全てのエージェントは既存手法である AMTDS および AMTDS/LD か、提案手法である AMTDS/LDPD によって目標選択戦略の決定および通信を行う。各目標選択戦略のパラメータは [4] と同一のものを使用している。

### 5.2 評価指標

継続的な清掃作業では、ごみの存在量と存在時間の両方を小さく抑えることが求められる。そのため、評価指標を以下の  $D_{t_s, t_e}$  で定義する。

$$D_{t_s, t_e} = \sum_{v \in V} \sum_{t=t_s+1}^{t_e} L_t(v). \quad (18)$$

ここで、 $t_s < t_e$  である。この  $D_{t_s, t_e}$  の値が小さいほど、清掃効率が良い。

### 5.3 結果と考察

環境 (a), (b) における、既存手法と提案手法の清掃効率の推移を図 2 に示す。図 2 より、両環境において提案手法である AMTDS/LDPD がごみの存在時間の総和を小さく抑え、高い効率を維持している。特に、より複雑な環境である環境 (b) においては、最終的に AMTDS に比べ 35.8%, AMTDS/LD に比べ 26.7% ごみの存在時間の総和を減少させている。この要

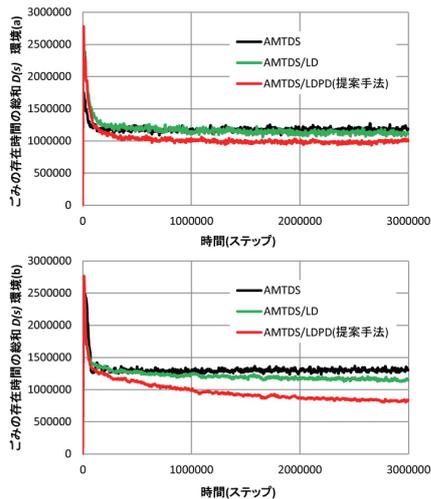


図 2: 清掃効率の推移比較

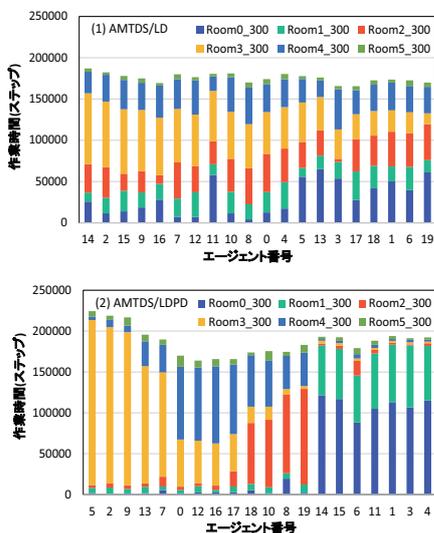


図 3: 環境 (b) における各 Room での作業時間比較

因を調べるため、環境 (b) において各エージェントが各 Room で作業をしていた時間を図 3 に示す。ここで、比較しやすくなるようエージェント番号は Room3 の作業時間の降順で並び替えてある。なお、この結果は無作為に選んだ 1 試行のものであるが、他の試行でも同様の現象が確認された。既存手法である AMTDS では全エージェントが事前に汚れやすさに関する正しい知識をもつため、作業時間の割合はほぼ同様であった。図 3(1) に示す AMTDS/LD では、学習によって間接的に他者がよく訪問する場所は汚れにくいと判断し分業を実現していたため (詳しくは [4] を参照されたい)、エージェント間で部屋ごとの作業時間にばらつきが発生している。提案手法である図 3(2) ではさらに作業時間の割合の差が顕著となり、その特徴は大きく分けて 4 種類に分かれている。1 つはエージェント番号 5, 2, 9, 13, 7 の Room3 を中心に作業するエージェント群である。Room3 は全部屋の中で最も汚れやすい面積が広く、非常に汚れやすい Region も存在する。この群は他のエージェントに比べ探索範囲を狭くしており、Room3 に作業を集中させたと言える。2 つめはエージェント番号 0, 12, 16, 17 の Room4 で

主として作業し、Room3 にも訪れている群である。Room4 は汚れやすい領域である Region が点在しているが、その面積は広くないため、Room3 にも訪れていると考えられる。3 つめはエージェント番号 18, 10, 8, 19 の Room2 で主として作業し、Room4 にも訪れている群である。この群は面積は狭いが汚れやすい Region が点在する Room2 と 4 を担当しており、競合が発生しやすい状況である。この群は前述の 2 つの群よりも広い探索範囲をもっており、競合を回避するために広い範囲を探索するようになったと考えられる。最後はその他の Room0 と Room1 で主に作業を行う群である。Room0, 1 は同じ汚れやすさのノードが広い範囲に分布している。そのため、競合が発生しにくく、広く巡回することが求められる空間である。また、この群はほぼ全ての試行において最も多いエージェント数を含んでいた。以上の結果より、提案手法はエージェントの分業を促進し、チーム編成を恣意的に行っていないにも関わらず競合のしやすさや環境の構造によって、似たような作業パターンをもつチームのような群を生み出すことが確認された。本提案手法は経路や目的地、戦略等の多様な内部情報の交換を行わず、重心ノード  $v_{center}$  と探索マップの重さ  $PDA_{sum}$  の比較のみによって通信を行う簡易なものだが、探索範囲の増減を組み合わせることで環境に合わせた分業を自然に実現したと言える。

## 6. 結論と今後の課題

本研究では複数エージェントの巡回清掃問題を題材に、簡易な通信によって分業を促進する手法を提案した。提案手法では重心と仕事量、探索範囲に着目し、これらの組み合わせによって既存手法よりも分業を促進し、複雑な環境では特に清掃効率を大きく向上させることを示した。また、エージェント間でチーム編成を恣意的に行わないにもかかわらず、提案手法によって環境に応じたチームのような群を形成できることも示唆した。今後の課題として、提案手法の汎用性を高めるために一様な環境でも応用できるような仕組みを導入する。

## 参考文献

- [1] M. Ahmadi and P. Stone. A multi-robot system for continuous area sweeping tasks. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pp. 1724 – 1729, 2006.
- [2] Y. Elor and A. M. Bruckstein. Multi-a(ge)nt graph patrolling and partitioning. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technologies*, pp. 52–57, 2009.
- [3] P.A. Sampaio, G. Ramalho, and P. Tedesco. The gravitational strategy for the timed patrolling. In *Tools with Artificial Intelligence (ICTAI), 2010 22nd IEEE International Conference on*, pp. 113–120, 2010.
- [4] A. Sugiyama and T. Sugawara. Meta-strategy for cooperative tasks with learning of environments in multi-agent continuous tasks. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pp. 494–500, 2015.
- [5] K. Yoneda, C. Kato, and T. Sugawara. Autonomous learning of target decision strategies without communications for continuous coordinated cleaning tasks. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, Vol. 2, pp. 216–223, Nov 2013.