

非通讯多 Agent 协作在 RoboCup 中的应用

郭 博,程家兴,张大强

(安徽大学 计算智能与信号处理教育部重点实验室,安徽 合肥 230039)

摘 要:智能体的行为预测是多智能体系统中的一个具有挑战性的问题。根据多 agent 系统(MAS)中的研究,为了提高进攻的速度,文中将一种新的合作策略引入到机器人足球中。首先,介绍了 MAS 的特点。同时,引入了一种 agent 削减算法来进行行为决策。进一步,调整了涉及到的 agent 的数目,在非通讯情况下实现了球员之间的协作。以后工作的重点是增加对对手的建模,以期提高进攻的成功率。

关键词:行为预测;非通讯;机器人足球;协作

中图分类号:TP242.6

文献标识码:A

文章编号:1005-3751(2006)04-0090-03

Non-communicative Multi-Agent Collaboration in RoboCup

GUO Bo, CHENG Jia-xing, ZHANG Da-qiang

(Ministry of Education Key Lab. of Intelligent Computing & Signal Processing at Anhui Univ., Hefei 230039, China)

Abstract: The prediction of the agent action in multi-agent system(MAS) has been a challenging problem. According to the research of multi-agent system, in order to improve speed of attack, a new strategy of collaboration is used in RoboCup simulation. First, introduce the characteristic of MAS. And an agent-minify arithmetic is used to do action decision-making. Furthermore, modify the number of agent involved, implement the collaboration between teammate in non-communication. In future work, it is important to use the modeling of opponent to increase the rate of good attack.

Key words: action prediction; non-communicative; RoboCup; collaboration

1 多 agent 系统

多 agent 系统由多个智能 agent 组成的,它一般具有个体行为独立自制、个体信息不完全、能力有限、无全局控制、数据分散化和计算异步等特点。多 agent 系统的数据和资源是分散的,每个 agent 对于所要完成的任务拥有不全面的信息或能力^[1]。多 agent 系统作为解决复杂系统问题的一个有效方法,能够利用并行分布式处理技术和模块化设计思想,把复杂系统划分成相对独立的 agent 子系统,通过 agent 之间的合作与竞争来完成对复杂问题的求解。

MAS 的体系结构是用于定义 MAS 体系结构的元素、体系结构元素之间的相互关系以及对体系结构元素的约束的一套规则^[2]。

●MAS 的体系结构有三种基本方案:

(1) 集中式结构(黑板结构):agent 之间通过集中式信息中心或公共资源共享区,进行相互通信、资源共享;

(2) 分散式结构:没有集中的信息交换中心或资源共享区,而是由各 agent 按照通信协议直接进行消息交换,

具有分散式协调通信;

(3) 混合式结构:是指集中与分散相结合的信息结构,可以采用的是阶梯式信息结构。

在多 agent 系统中,每个自主 agent 拥有解决问题的不完全的信息和能力。agent 系统要通过相互协作进行复杂问题求解,就需要一定的协调机制实现 agent 之间的任务分派和冲突消解。多 agent 合作问题求解是 Multi-agent 系统的重要研究领域,目前的研究工作主要集中在任务分配、agent 通讯机制、冲突消解策略等方面^[3]。

●多 agent 系统具有以下特点:

(1) 分布性。MAS 不仅在结构上是分布的,在逻辑上也是分布的,其中的 agent 具有不完全的知识和分布的决策能力,计算也是异步进行的;

(2) 开放性。agent 无论从概念上还是从实现上都是—种封装模型,其内部结构和算法可以由不同人在不同时间和地点采取不同方法加以实现;

(3) 适应性。对于环境的变化和不确定性因素,agent 可以通过在协调机制下交互和自学习适应新的条件,保持系统的鲁棒性。

2 多 agent 非通讯协作

多个 agent 之间完成协作可以通过以下几个不同的方法^[4]:通讯、学习或协议。首先,如果利用通讯,agent 需

收稿日期:2005-07-31

作者简介:郭 博(1981-),男,安徽阜阳人,硕士研究生,研究方向为人工智能理论、智能软件设计及工程优化;程家兴,博士,教授,博导,研究方向为智能计算、算法分析及设计及最优化方法。

要得到其他 agent 的行为,而且限制其他 agent 的选择来简化协作。第二,可以重复进行系统学习。每个 agent 利用先前的交互对其他 agent 的行为进行预测,同时选择自身的动作。最后,可以利用协议来约束 agent 的动作选择。利用此协议可以在所有可能的平衡点选出一个最优点。一旦所有 agent 都遵从这个协议,系统可以得到好处。这种通用的、领域独立的方法会得到一个最优连接动作,而且在执行中 agent 不必直接协商它们的动作。假定所有的平衡点可以被找到,而且协商是基于相同的平衡点每个 agent 选择自己的行为的结果。不过,连接动作的数量随着 agent 的数目呈指数增长,所以在许多 agent 的情况下确定所有平衡点是不可行的。

2.1 削减 agent 的算法

MAS 可以由 $\langle n, A_1, \dots, A_n, R_1, \dots, R_n \rangle$ 来表示,其中 n 是 agent 的数目, A_i 是第 i 个 agent 的动作集, R_i 是第 i 个 agent 的决策函数。其中的决策函数通过动作集映射到实数集上,可用得到的实数大小来判断决策的好坏。每个 agent 自主地从允许的动作集中选择动作,然后根据其他 agent 选择的动作得到决策值。而全局决策函数 $R(a)$ 可以分解成一个局部决策函数(包括较少的 agent)的线性组合,即若干函数的代数和。

为了解决协作问题并找到最优连接动作,引入了一个 agent 消减算法,类似在 Bayesian 网络中进行变量消减。算法如下:

- (1) 选择一个和其他 agent 联系最少的 agent;
- (2) 收集和它相关 agent 的决策函数;
- (3) 根据相关 agent 可能的动作有条件地优化自己的决定并反馈给它们。这个局部决策只与此 agent 有关,因此这个 agent 可以被消减到;
- (4) 不断重复(1)~(3),直到只剩最后一个 agent;
- (5) 最后得到一个 agent 消减顺序;
- (6) 按照以上削减顺序的逆序方式,agent 逐一选择最优动作(以最大化最终决策为目的)。

算法结束。

2.2 非通讯的 MAS

在很多特定的动态情况下(比如 RoboCup), agent 会由于失效和时间限制导致不能和其他 agent 通讯^[5]。而上述削减算法需要 agent 知道周边 agent 的决策函数值,并且又要将自己的决策函数反馈给其他 agent。相似的,在逆序的过程中也要和其他 agent 交流彼此的动作决定。这样的话,上述削减算法就很可能无法在 RoboCup 中运用。

不过,如果能让 agent 知道其他 agent 的决策函数,无需彼此通讯,就可以继续使用此算法。为此,需要下边的假设:其一,所有 agent 的决策函数是共有知识;其二,每个 agent 都可以预测其他 agent 的动作。

同时,因为 agent 之间没有通讯,每个 agent 需要自主地预测其他 agent 的动作以确定自己的动作。这样,每个

agent 的计算开销会明显增加。而在通讯的情况下,每个 agent 自身的计算不多,但用于通讯的计算开销较大。相比之下,agent 之间没有通讯所需附加的计算是可以并行进行,从而多 agent 系统从主体上缩短了时间。同时,在非通讯的过程中的削减次序无须事先确定,每个 agent 可以自由选择 and 快速选择自身最佳动作的削减次序(其中任何一个具体的削减次序只会影响算法的速度,并不会影响计算出最佳的动作)。

3 在 RoboCup 中实现

在无通讯的 RoboCup 中,如果所有 agent 都得到同一的情况,共同知识可以被保证。而实际情况下,agent 对环境的观察是不完全的(比如在 RoboCup 中 agent 的视野就是受限的)^[4]。同时对于有些情况下的状态是 agent 自身的大致估计。在 RoboCup 中,agent 需要一组队友的坐标,而且同时要知道对手的位置,还有对球的位置的大致估计(如只知道球离自己很远)。

3.1 具体分析

在理想情况下,应该考虑全场(10 名球员)的位置和动作,按照上述的削减算法逐一确定其行为。在这种情况下,为了让每个球员能够预测场上其他所有队友的行为,不得不在每个 agent 的决策函数里对其他 9 名球员一一建模。这样,每个球员会被其他球员创建出 9 个模型,而这些模型之间没有任何沟通(通讯)。由于 agent 的自主性,很难预测出 agent 的最终决策。实验表明,对同一 agent 行为的预测的结果很难保持一致,更多的情况是出入较大,甚至相反(一个 agent 预测队友向前跑,而另一个预测同一队友向后跑)。

一个 agent 本来只要对自身的行为进行决策,现在却需要同时也对其他 9 个队友的行为进行预测。这就相当于一个 agent 要同时作出 10 个不同情况下的决策。很不幸,这种情况下,agent 的计算量大大增加了,因而不利于 agent 在接收到最新信息后快速作出决策。更糟的是,由于时间的问题,agent 有时错过向系统发送信息的机会,导致不能进行正常的决策。

在上述情况下,球员之间的“默契”没能切实实现,同时由于决策时间的滞后,进攻的速度根本得不到提高,反而有时会显得杂乱无章,这和他的原意恰恰相反。所以同时考虑所有的球员的做法是不太可行的。

3.2 精减要素

基于上文的讨论,要把一个 agent 所考虑的其他 agent 的数目降到最少。考虑到比赛的具体情况(一个周期里每个 agent 最多执行一个动作,也就是一个决策),最后把 agent 的数目定为 2 到 3 个——其中包括一名持球队员和 1、2 名无球队员。

为了简化,只考虑离球最近的 2、3 名球员的行为(在这里假定只有离球最近的球员参与进攻)。显然,很容易把球员的决策分为两大类:

1) 持球队员决策。

对于持球队员来说,首先考虑的是把球控制在自己的范围内,如果球被对方球员截到,进攻就无从谈起。在对手离球较近(其威胁性也大)的情况下,持球队员优先执行各种控球的策略,对球进行彻底的控制。而在对手离球较远(其威胁性也小)的情况下,持球队员才可以考虑如何组织进攻。组织进攻包括自己是否应该继续带球、预测周边的队友下一步的动作(以便更好地传球)及其传球的路线选择等。持球队员的决策在一个完整的进攻中处于核心地位(可以直接影响球的位置和速度),同时可以对无球队员决策进行方向性的指导。好的持球队员的决策决定了进攻的成败。持球队员是进攻的发起者。

2) 无球队员决策。

对于无球队员来说,无须考虑控球的问题,也暂时无法影响球的位置和速度。无球队员的决策是以对持球队员行为的预测为核心。无球队员根据预测出持球队员动作(带球或传球)的情况,决定自己的行为。无球队员的行为主要是跑位,根据设定的有利进攻点来调整自身的方向。同时无球队员应做好对方可能会断球、球被断后进行快速抢球及其完成从无球队员到持球队员转换的准备。无球队员是进攻的响应者。

如图 1(a)所示,持球队员首先预测了队友的动作(图中虚线箭头所示),将球沿实线箭头传给队友;同时如图 1(b)所示,无球队员首先预测了队友的动作(传球),并预测球的路线(图中虚线箭头所示),同时决定沿实线箭头跑位。比较同一情况下两名不同位置的球员的决策,可以发现虽然预测和实际的动作有出入,但从总体上可以认为这两个 agent 完成了配合,达到了想要的结果(实现了非通讯状态下的球员之间的合作,即球员之间产生了“默契”)。

实验表明,这种以持球队员为核心的进攻策略简化了 agent 考虑的因素,提高了传球的目的性,加快了进攻的速

度,从而实现了预期的球员之间的“默契”。

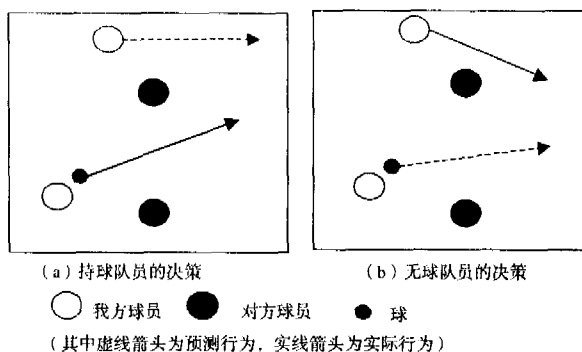


图 1 同一情况下球员的决策

4 结束语

从多 agent 系统研究入手,根据 RoboCup 的具体情况,实现了非通讯状态下 agent 之间的配合。在调整了考虑的 agent 的数目后,我们的策略达到了预期的效果。下一步,考虑增加对对手的建模,提高传接球的质量,以期得到更好的进攻成功率。

参考文献:

- [1] 咸鹤群,孟庆春,殷波,等.多 Agent 系统中潜在角色值研究[J].哈尔滨工业大学学报,2003,35(9):1089-1092.
- [2] 祁正华,任勋益.基于 MAS 的智能决策支持系统研究[J].微机发展,2004,14(9):14-16.
- [3] 陈进才.多 Agent 系统的形式化理论研究[D].西安:西安交通大学,2000.
- [4] Kok J R, Spaan M T J, Vlassis N. Non-communicative multi-robot coordination in dynamic environments[J]. Robotics and Autonomous Systems, 2005, 50: 99-114.
- [5] 张颖霞,杨宜民,陈波,等.多智能体团队合作在机器人足球赛中的应用[J].微机发展,2004,14(7):112-114.

(上接第 89 页)

中,不得不存储实施过程中产生的各种状态,而造成空间复杂度比其它算法要高。

文中选择动态规划算法来解决背包问题,且认为它在空间上是可以承受的,而采用搜索算法在时间上却是难以承受的。舍空间而取时间,这正是该算法在解决背包问题上体现出的一种优越性。

5 结束语

文中把动态规划方法用于求解背包问题,其算法是以空间换时间来解决问题的一种有效的方法。在解决实际问题的过程中,面对复杂问题而分解出的子问题往往并不是孤立的,要解决问题的子问题常常需要调用相同的子问题的子问题。如果它们之间也不是独立的,采用分治法势必会在解决同样的问题上造成很大的时间浪费。由于动态规划方法对一个问题或多次出现的相同问题仅需解决

一次,因此可节省大量的计算时间而具有更高的运行效率。

参考文献:

- [1] 玄光男,程润伟.遗传算法与工程优化[M].北京:清华大学出版社,2004.55-63.
- [2] 罗小虎,赵雷.一个解决 0/1 背包问题的蚁群算法[J].苏州大学学报(工科版),2004(2):41-44.
- [3] Supowit K. Finding a Maximum Planar Subset of a Set of Nets in a Channel[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and systems, 1987, 6(1): 93-94.
- [4] 阮沈勇,王永利,桑群芳. Matlab 程序设计[M].北京:电子工业出版社,2004.
- [5] 施益昌,郑贤斌,李自立.基于 Matlab 动态规划中最短路线的实现程序[J].电脑学习,2003(12):38-40.