第5卷第3期 软 件 学 报 Vol. 5, No. 3
1994年3月 JOURNAL OF SOFTWARE Mar. 1994

广义 AND/OR 图的自底向上的启发式搜索算法 BHAO* *

王士同

(镇江船舶学院计算机系,镇江 212003)

摘要 本文首先根据三角模概念,定义了一类新的更具普遍意义的广义 AND/OR 图.根据新定义的启发式函数 h(n,x)以及广义 AND/OR 图的最佳解树之所有子树亦是最佳子解树的原理,提出了广义 AND/OR 图的自底向上的启发式搜索算法 BHAO*.文中证明了算法 BHAO* 的可采纳性.本文还提出了两类新的启发式函数的单调限制概念,并据此研究了算法 BHAO* 的单调限制性质,研究了两个 BHAO* 算法间的比较性质.

关键词 广义 AND/OR 图,启发式搜索,启发式函数,单调限制,算法.

AND/OR 图的启发式搜索技术是人工智能启发式搜索技术中的重要内容,有着广泛的直接应用.目前已出现了不少 AND/OR 图的启发式搜索算法,如 AO*,AO$_\varepsilon^*$,DPAO*,NAO*[2]等.在客观世界中,上述算法所基于的 AND/OR 图在描述问题求解时还有许多局限性.本文依据三角模的概念,定义一类新的更具普遍意义的广义 AND/OR 图.和 Nilsson 教授定义的 AND/OR 图的启发式函数 h(n)不同,本文新定义了广义 AND/OR 图的启发式函数 h(n,x);根据广义 AND/OR 图的最佳解树其所有子解树亦是子最佳解树的原理,提出了广义 AND/OR 图的自底向上的启发式搜索算法 BHAO*(Backward Heuristic Search Algorithm for General AND/OR Graph).算法 BHAO* 是可采纳的,即若存在解树,BHAO* 一定能找到最佳解树.本文定义了新的启发式函数的1型和2型单调限制,研究了算法 BHAO* 的单调限制性质.当两个算法 BHAO* 所用的启发式函数满足1型单调限制时,本文还证明了两个 BHAO* 算法间的比较定理.本文的结果有助于人工智能启发式搜索技术的深入研究.

1 广义 AND/OR 图

定义1. 三角模 S 和 T 见文献[9].

这里特别要指出的是,由三角模的定义知模 S 和模 T 是单调非减的.

---

* 本文1991年3月24日收到,1991年6月23日定稿

本研究是国家自然科学基金资助项目.作者王士同,31岁,副教授,主要研究领域为人工智能,模糊数学,神经网络.

本文通讯联系人:王士同,镇江 212003,镇江船舶学院计算机系

定义 2. 连续三角模$\bigwedge_{i=1}^{k} t$是指如下的运算：

$$\bigwedge_{i=1}^{k} t(a_1,a_2,\cdots,a_k)=t(a_1,t(a_2,\cdots,a_k))=t(a_1,t(a_2,t(a_3,\cdots,a_k)))$$
$$=\cdots\cdots=t(a_1,t(a_2,t(a_3,\cdots,t(a_{k-1},a_k))))$$

容易推出，连续三角模是单调非减的.

定义 3. 广义 AND/OR 图是如下的超图 G：(1)此超图中的每个节点表示一个问题，root(G)表示超图的根，是要解的原问题. 一个问题转化成若干子问题可通过超弧来描述. 一个超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$ 就表示问题 n 可通过求解子问题 $n_1,n_2,\cdots,n_k$ 来解决. 没有后继的节点称之为端节点(端节点的耗散值约定均大于 0)，否则称之为非端节点. (2)对超图 G 中的端节点 n，c(n)表示求解问题 n 所需的耗散值. 对超图 G 中的每一超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，连续三角模 t 运算被用来计算求解 n 所须的耗散值，即为$\bigwedge_{i=1}^{k} t(v_1,v_2,\cdots,v_k)$，其中 $v_1,v_2,\cdots,v_k$ 表示求解问题 $n_1,n_2,\cdots,n_k$ 所须的耗散值.

定义 4. 广义 AND/OR 图 G 的解树 T_r 定义为：(1)T_r 是 G 的子树；(2)T_r 的根节点是 G 的根节点；(3)若广义 AND/OR 图 G 的一个非端节点 $n\in T_r$，则必存在唯一的一条超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，使得此超弧 $p\in T_r$(注意：p 仅是 G 中从 n 出发的超弧之一)；(4)解树 T_r 的耗散值计算如下述. 设 $C_{T_r}(n)$表示在解树 T_r 中求解问题节点 n 所须的耗散值，则若 n 是端节点，就定义 $C_{T_r}(n)=c(n)$；若 n 是非端节点且在 T_r 中存在一超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，则 $C_{T_r}(n)=\bigwedge_{i=1}^{k} t(C_{T_r}(n_1),C_{T_r}(n_2),\cdots,C_{T_r}(n_k))$.

下文中还设定 $c^*(n)$表示在广义 AND/OR 图 G 中根节点为 n 的最佳解树的耗散值. 所谓最佳解树就是具有最小耗散值的解树.

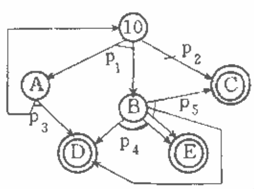

图1 一个广义AND/OR图

例如：一个广义 AND/OR 图及其耗散值函数如下述. 图中双圆圈之节点表示端节点.

超弧的耗散函数：

$$t_{p_1}(x,y)=S_\infty(x,y)=\min(1,x+y)$$

$$t_{p_2}(x)=T_1(0.5,x)=\frac{1}{2}x$$

$$t_{p_3}(x,y)=T_3(x,y)=\frac{x\cdot y}{1+(1-x)(1-y)}$$

$$t_{p_4}(x,y)=S_0(x,y)=\max(x,y)$$

$$t_{p_5}(x,y,z)=\bigwedge_{i=1}^{3} T_0(x,y,z)=\min(x,y,z)$$

端节点的耗散值：

$C(C)=0.7$，$C(D)=0.3$，$C(E)=0.5$

图 2 给出了图 1 所示 AND/OR 图的一个解树，以及在此解树中诸耗散值之计算过程. 注意，从图 1 可易知，图 2 中的 S_1 节点就是 S 节点，如此标记仅是为了计算时不易混淆.

$C_{T_r}(C)=C(C)=0.7, C_{T_r}(D)=C(D)=0.3$

$C_{T_r}(E)=C(E)=0.5$

$C_{T_r}(S_1)=t_{p_2}(C_{T_r}(C))$

$$=\frac{1}{2}\times 0.7=0.35$$

$$C_{T_r}(A)=t_{p_3}(C_{T_r}(S_1),C_{T_r}(D))$$
$$=t_{p_3}(0.35,0.2)$$
$$=\frac{0.35\times 0.3}{1+0.65\times 0.7}=0.0721$$
$$C_{T_r}(B)=t_{p_5}(C_{T_r}(C),C_{T_r}(D),C_{T_r}(E))$$
$$=\bigwedge_{i=1}^{3}T_0(0.7,0.3,0.5)$$
$$=\min(0.7,0.3,0.5)=0.3$$
$$C_{T_r}(S)=t_{p_1}(C_{T_r}(A),C_{T_r}(B))$$
$$=t_{p_1}(0.0721,0.3)$$
$$=\min(1,0.0721+0.3)=0.3721$$

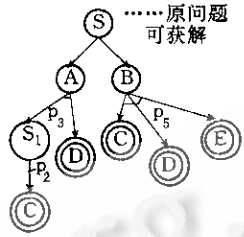

图 2　一个解树及其耗散值计算过程

从前述定义以及上例可以看出，与以前所见的 AND/OR 图相比，本文所定义的广义 AND/OR 图更具普遍性，更具实用意义.

定理 1. 对广义 AND/OR 图 G 的每个节点 n，恒有下列递归方程成立：(1)若 n 是端节点，则 $c^*(n)=c(n)$；(2)若 n 是非端节点，则

$$c^*(n)=\min\{\bigwedge_{i=1}^{k}t(c^*(n_1),c^*(n_2),\cdots,c^*(n_k))\mid 存在超弧\ p:n\rightarrow n_1,n_2,\cdots,n_k\}$$

证明：(1)若节点 n 是端节点，则定理显然成立.

(2)设存在一个子问题节点 n_i，它的非最佳耗散值 $c(n_i)$ 能使得问题节点 n 获得最佳解，即

$$c^*(n)=\bigwedge_{i=1}^{k}t(c^*(n_1),\cdots,c(n_i),\cdots,c^*(n_k))$$

但因 $c(n_i)\geqslant c^*(n_i)$，又 $\bigwedge_{i=1}^{k}t$ 是单调非减的，故

$$\bigwedge_{i=1}^{k}t(c^*(n_1),\cdots,c(n_i),\cdots,c^*(n_k))\geqslant\bigwedge_{i=1}^{k}t(c^*(n_1),\cdots,c^*(n_i),\cdots,c^*(n_k))$$

这与假设矛盾，故本定理成立.

定理 2. 对广义 AND/OR 图 G 的每一个节点 n，若存在一个根节点为 n 的最佳解树 $T_r(n)$，则 $T_r(n)$，则 $T_r(n)$ 的所有子树亦是最佳解树.

定理 2 可以从定理 1 直接推得，定理 2 很重要，它是设计广义 AND/OR 自底向上的启发式搜索算法 BHAO* 的思想基础.

2　自底向上的启发式搜索算法 BHAO*

和以前常见的有关 AND/OR 图的启发式函数不同，本文新定义一个独特的启发式函数.

定义 5. 设 n 是广义 AND/OR 图的一个节点，x 是根节点为 n 的某棵解树 T 的耗散值，令

$$h^*(n,x)=\min\{C_{T_r}(root(G))\mid C_{T_r}(root(G))是广义\ AND/OR\ 图\ G\ 的某解树\ T_1\ 的耗散值，且\ T\ 是\ T_1\ 的子树\}$$

且 $h^*(\mathrm{root}(G),c^*(\mathrm{root}(G)))=c^*(\mathrm{root}(G))$. 启发式函数 $h(n,x)$ 是 $h^*(n,x)$ 的估计，且 $h(\mathrm{root}(G),x)=x$.

容易看出，$h^*(n,x)$ 表示的是广义 AND/OR 图 G 在包含子解树 T 情况下之最佳解树的耗散值. 启发式函数 $h(n,x)$ 与常见的启发式函数 $h(n)$ 其形式和含义是有本质区别的.

下面提出广义 AND/OR 图的自底向上的启发式搜索算法 BHAO.

算法 BHAO：

(1)初始时 OPEN 表为空表，CLOSED 表为广义 AND/OR 图的所有端节点集合. 对任意节点 n，若 $n\in$ CLOSED，则置 $\mathrm{val}(n)\leftarrow c(n)$.

(2)对任意节点 n，若 $n\notin$ (OPEN 或 CLOSED)，且存在一超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，使得 $\{n_1,n_2,\cdots,n_k\}\subseteq$ CLOSED，则 OPEN$\leftarrow$OPEN$\cup\{n\}$

同时计算：

$$\mathrm{val}(n)=\min\{\bigwedge_{i=1}^{k}t(\mathrm{val}(n_1),\mathrm{val}(n_2),\cdots,\mathrm{val}(n_k))\mid \text{存在一超弧 } p:n\rightarrow n_1,n_2,\cdots,n_k,\text{使得 }\{n_1,n_2,\cdots,n_k\}\subseteq \mathrm{CLOSED}\}$$

(3)对任意节点 n，若 $n\in$ CLOSED，且存在一超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，使得 $\{n_1,n_2,\cdots,n_k\}\subseteq$ CLOSED，且 $\mathrm{val}(n)>\bigwedge_{i=1}^{k}t(\mathrm{val}(n_1),\cdots,\mathrm{val}(n_k))$，则重新计算：

$$\mathrm{val}(n)=\min\{\bigwedge_{i=1}^{k}t(\mathrm{val}(n_1),\cdots,\mathrm{val}(n_k))\mid \text{存在一超弧 } p:n\rightarrow n_1,n_2,\cdots,n_k,\text{使得 }\{n_1,n_2,\cdots,n_k\}\subseteq \mathrm{CLOSED}\}$$

同时：OPEN$\leftarrow$OPEN$\cup\{n\}$，CLOSED$\leftarrow$CLOSED$-\{n\}$

(4)对任意节点 n，若 $n\in$ OPEN，且存在一超弧 $p:n\rightarrow n_1,n_2,\cdots,n_k$，使得 $\{n_1,n_2,\cdots,n_k\}\subseteq$ CLOSED，且 $\mathrm{val}(n)>\bigwedge_{i=1}^{k}t(\mathrm{val}(n_1),\cdots,\mathrm{val}(n_k))$，则重新计算：

$$\mathrm{val}(n)=\min\{\bigwedge_{i=1}^{k}t(\mathrm{val}(n_1),\cdots,\mathrm{val}(n_k))\mid \text{存在一超弧 } p:n\rightarrow n_1,n_2,\cdots,n_k,\text{使得 }\{n_1,n_2,\cdots,n_k\}\subseteq \mathrm{CLOSED}\}$$

(5)若 OPEN 表为空，则失败退出.

(6)对 OPEN 表按 $h(n,\mathrm{val}(n))$ 大小升序排序. 若 OPEN 表中的第一个元素 m 即第一个节点 m 就是 root(G)，则算法成功结束，且最佳解树的耗散值就是此时的 val(root(G)). 否则，OPEN$\leftarrow$OPEN$-\{m\}$，CLOSED$\leftarrow$CLOSED$\cup\{m\}$.

(7)GOTO 步骤(2).

和启发式搜索算法 RA^*[2]、A^*[6]类似，在广义 AND/OR 图的启发式搜索算法 BHAO 中，也使用了 OPEN、CLOSED 两个表. 从算法 BHAO 中的步骤(1)、(2)可以看出，广义 AND/OR 图 G 的某节点 n 属于 OPEN 或 CLOSED 表，当且仅当至少存在一个根节点为 n 的解树，此解树的所有节点均已属于 CLOSED 表. 对于 OPEN 或 CLOSED 表中的某节点 n，val(n)则表示根节点为 n 的一解树的耗散值. 根据算法步骤(2)—(4)，以及连续三角模的单调非减性质，容易看出，对于 OPEN 或 CLOSED 表中的所有节点 n，均有：

$$\mathrm{val}(n)\leqslant\min\{C_{T_r}(n)\mid C_{T_r}(n)\text{是根节点为 } n \text{ 的解树 } T \text{ 的耗散值，且解树 } T \text{ 中的除 } n \text{ 外的所有节点均已属于 CLOSED 表}\}.$$

定义 6. 若对广义 AND/OR 图 G 中的所有节点 n，启发式函数 $h(n,x)$ 均满足 $h(n,x) \leqslant h^*(n,x)$，则启发式算法 BHAO 此时就称启发式算法 BHAO*.

定理 3. 算法 BHAO* 结束前的任何时刻，OPEN 表上存在一个节点 n'，它是在广义 AND/OR 图 G 的最佳解树上，且具有 $h(n',val(n')) \leqslant c^*(root(G))$.

证明：设排序的序列 $(n_0, n_1, \cdots, root(G))$ 是按超弧连接构成的广义 AND/OR 图 G 的一个最佳解树（可能有多个），则对算法 BHAO* 结束前的任何时刻，令 n' 是 OPEN 表上这一序列中的第一个节点（至少必须有这样的节点，因为算法一开始，所有端节点均在 CLOSED 表中，根据算法步骤(2)，所有端节点的父节点均在 OPEN 表上，而且算法在选择到 root(G) 节点时会结束）. 注意，由于节点 n' 处在广义 AND/OR 图 G 的一个最佳解树中，故根据定理 2，有

$$h(n',val(n')) = h(n',c^*(n')) \quad ; \quad val(n') = c^*(n')$$

根据定义 6 知算法 BHAO* 满足：$h(n',c^*(n')) \leqslant h^*(n',c^*(n'))$.

由于最佳解树中的任一节点 n 的 $h^*(n,c^*(n))$ 均等于 $h^*(root(G),c^*(root(G)))$，即等于 $c^*(root(G))$，故有

$$h(n',val(n')) = h(n',c^*(n')) \leqslant h^*(n',c^*(n')) = c^*(root(G))$$

进而知本定理成立.

定理 4. 启发式算法 BHAO* 是可采纳的，即若广义 AND/OR 图存在最佳解树，则算法 BHAO* 一定将由于找到一最佳解树而结束.

证明：(1)首先证明算法 BHAO* 一定会找到一个解树而结束.

分析算法 BHAO，可以看出算法要么在算法步骤(5)结束，要么在算法步骤(6)结束. 但如果广义 AND/OR 图存在解树，结束前 OPEN 表永远不至于空. 根据定理 3，将总有一个节点在 OPEN 表上，且在最佳解树中，故算法一定会找到一个解树而结束.

(2)其次证明算法一定将由于找到最佳解树而结束.

假定算法 BHAO* 未找到一最佳解树即找到一非最佳解树 T 而结束，即 $C^T_{T_r}(root(G)) > C^*(root(G))$，其中 $C^T_{T_r}(root(G))$ 表示非最佳解树 T 的耗散值. 根据启发式函数 $h(n,x)$ 的定义知：

$$h(root(G)),C^T_{T_r}(root(G))) = C^T_{T_r}(root(G)) > c^*(root(G))$$

但是根据定理 3 知，算法 BHAO* 结束前在 OPEN 表上存在一节点 n'，且 $h(n',val(n')) < c^*(root(G))$，又 n' 在一最佳解树中，故有：

$$h(n'val(n')) < h(root(G),C^T_{T_r}(root(G)))$$

因此，此时算法 BHAO* 将选节点 n' 扩展而不选根节点 root(G). 这和假设相左. 综合以上两步证明知算法 BHAO* 是可采纳的.

定理 5. 算法 BHAO* 选来扩展的任一节点 n，均有：$h(n,val(n)) \leqslant c^*(root(G))$.

证明：令节点 n 是算法 BHAO* 选来扩展的任一节点. 若 $n = root(G)$，由定理 4 及算法 BHAO* 的步骤知 $h(root(G)),val(root(G)) = c^*(root(G))$，故假定 n 不是 root(G). 现在 BHAO* 结束前选了节点 n，故据定理 3，OPEN 表上必存在过某节点 n'，它在广义 AND/OR 图的一最佳解树中，且 $h(n'val(n')) \leqslant c^*(root(G))$. 若 $n = n'$，则本定理成立. 否则，算法

BHAO* 选 n 扩展而不选 n'，故有：

$$h(n,val(n))<h(n',val(n'))\leqslant c^*(root(G))$$

i.e　　$$h(n,val(n))\leqslant c^*(root(G))$$

进而知本定理成立.

3　算法的单限制及算法间的比较性质

本节首先定义 2 类启发式函数之单调限制，然后研究算法 BHAO* 的单调限制性质和算法的比较性质. 注意，这里定义的单调限制与常见的启发式函数之单调限制是不同的.

定义 7. 广义 AND/OR 图的启发式函数 $h(n,x)$ 称为 1 型单调限制，若 $\forall\ x_1,x_2\in R$，且 $x_1>x_2$，则 $h(n_1,x_1)>h(n_1,x_2)$.

定义 8. 广义 AND/OR 图的启发式函数 $h(n,x)$ 称为 2 型单调限制，若对根节点为 n_1 且耗散值为 $C_{T_r}^{T_1}(n_1)$ 的解树 T_1，根节点为 n_2 且耗散值为 $C_{T_r}^{T_2}(n_2)$ 的解树 T_2，又 T_2 是 T_1 的子树，则有：

$$h(n_1,C_{T_r}^{T_1}(n_1))\geqslant h(n_2,C_{T_r}^{T_2}(n_2))$$

定理 6. 若广义 AND/OR 图 G 的启发式函数 $h(n,x)$ 是 1 型和 2 型单调限制的，则自底向上的启发式搜索算法 BHAO* 对任选扩展的节点 n，早已找到一最佳解树，即 $val(n)=c^*(n)$.

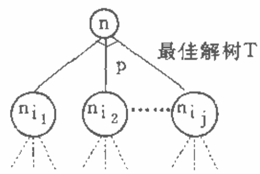

图3

证明：设节点 n 是算法 BHAO* 要扩展的任一节点. 若 $n=root(G)$，则算法 BHAO* 此时成功结束，且 $val(n)$ 就是广义 AND/OR 图的一最佳解树之耗散值. 若 $n\neq root(G)$，设序列 $n_0,n_1,\cdots,n$ 是从端节点按超弧连接起来的根节点为 n 的最佳解树 T，不妨设 T 如图 3 所示. 令节点 $n_{i_{j-1}}$ 是算法 BHAO* 选节点 n 扩展时属于 CLOSED 表中的解树 T 中的最末一个节点，而节点 n 因刚被选作扩展而不属于 CLOSED 表，故最佳解树 T 中的节点 n_{i_j} 在算法 BHAO* 选节点 n 扩展时在 OPEN 表上，因而根据算法 BHAO* 可知，必有：

$$h(n,val(n))\leqslant h(n_{i_j},val(n_{i_j}))\tag{1}$$

由于以根节点 n_{i_j} 的解树亦是最佳子解树(根据定理 2)，故有：

$$val(n_{i_j})=c^*(n_{i_j})$$

$$h(n,val(n))\leqslant h(n_{i_j},c^*(n_{i_j}))\tag{2}$$

根据以根 n_{i_j} 的子最佳解树是以根 n 的最佳解树之子树，又由于启发式函数 $h(n,x)$ 满足 2 型单调限制，故有：

$$h(n_{i_j},c^*(n_{i_j}))\leqslant h(n,c^*(n))\tag{3}$$

根据式(2)和式(3)，有　$h(n,val(n))\leqslant h(n,c^*(n))$.

又由于启发式函数 $h(n,x)$ 满足 1 型单调限制，故从前式可导出：

$$val(n)\leqslant c^*(n)\tag{4}$$

很容易想见，恒有下式成立：$val(n)\geqslant c^*(n)$. 故根据此式和式(4)可推知：$val(n)$

$=c^*(n)$，进而知本定理成立.

定理 6 很重要，它揭示了若启发式函数 $h(n,x)$ 满足 1 型和 2 型单调限制，则广义 AND/OR 图的自底向上的启发式搜索算法 BHAO* 中的步骤(3)可直接省略.

定义 9. 对两个算法 BHAO*，即 $BHAO_1^*$ 和 $BHAO_2^*$，称算法 $BHAO_2^*$ 比算法 $BHAO_1^*$ 有较多的信息，若各自所使用的广义 AND/OR 图的启发式函数 $h^1(n,x)$，$h^2(n,x)$ 满足：$h^1(n,x)<h^2(n,x)$.

定理 7. 对两个 BHAO* 算法 $BHAO_1^*$，$BHAO_2^*$，若 $BHAO_2^*$ 比 $BHAO_1^*$ 的信息多，且各自的启发式函数满足 1 型单调限制，则在任一具有解树的广义 AND/OR 图 G 上，$BHAO_1^*$ 所扩展的节点数至少和 $BHAO_2^*$ 所扩展的节点数一样多.

证明：对算法 $BHAO_2^*$ 所得的广义 AND/OR 图 G 的解树之结束处的节点高度用归纳法证明本定理.

(1)归纳基础：若 $BHAO_2^*$ 扩展的解树中节点 n 的高度为 1，则 $BHAO_1^*$ 一定也这样，但此时 n 仅是端节点的父节点. 若 n 是 root(G)，则两算法均不必再扩展此节点，即算法结束. 若 n 不是 root(G)，则两算法均须自底向上地扩展节点 n.

(2)设 $BHAO_1^*$ 扩展了 $BHAO_2^*$ 所扩展的解树中的高度为 k 或小于 k 的全部节点，现在就需证明 $BHAO_2^*$ 所扩展的解树中的高度为 k+1 的任意节点也可由 $BHAO_1^*$ 加以扩展.

根据归纳法假设，$BHAO_2^*$ 所得的解树中的节点 n 的任一后裔节点都可由 $BHAO_1^*$ 扩展，故有：

$$val^{BHAO_2^*}(n)\geq val^{BHAO_1^*}(n) \tag{5}$$

上述断言可根据算法步骤以及连续三角模 $\bigwedge_{i=1}^{k} t$ 的单调非减性和下式：

$$val(n)=\min\{\bigwedge_{i=1}^{k} t(val(n_1),\cdots,val(n_k))\mid \text{存在超弧 } p:n\rightarrow n_1,n_2,\cdots,n_k,\text{使得}\{n_1,n_2,\cdots,n_k\}\subseteq CLOSED\}$$

极容易推断出.

假定算法 $BHAO_1^*$ 不能扩展由 $BHAO_2^*$ 的节点 n. 在 $BHAO_1^*$ 结束时，n 必须在 OPEN 表上，因为 $BHAO_1^*$ 已扩展了 n 的后裔节点. 由于 $BHAO_1^*$ 未扩展节点 n 而在最佳解树上结束，故有：

$$h^1(n,val^{BHAO_1^*}(n))>h^1(root(G),c^*(root(G)))=c^*(root(G)) \tag{6}$$

根据 $h^1(n,x)$的 1 型单调限制性质，从式(5)和式(6)可得：

$$h^1(n,val^{BHAO_2^*}(n))>c^*(root(G)) \tag{7}$$

根据定理 5，当算法 $BHAO_2^*$ 扩展节点 n 时恒有：

$$h^2(n,val^{BHAO_2^*}(n))\leqslant c^*(root(G)) \tag{8}$$

根据式(7)和式(8)，可得

$$h^2(n,val^{BHAO_2^*}(n))<h^1(n,val^{BHAO_2^*}(n))$$

这一结果与假定 $h^2(n,x)>h^1(n,x)$相左，故 $BHAO_1^*$ 必扩展由 $BHAO_2^*$ 所扩展的节点 n. 根据 n 的任意性知本定理成立.

结束语：本文定义了一类更加广义的 AND/OR 图，提出了独特的启发式函数以及其单

调限制概念.根据自底向上的搜索思想，提出了广义 AND/OR 图的启发式搜索算法 BHAO*.算法 BHAO* 是可采纳的.当算法 BHAO* 的启发式函数 h(n,x)满足 1 型和 2 型单调限制时，BHAO* 算法还可进一步简化.文中还研究了算法 BHAO* 的比较性质.笔者深信，本文对广义 AND/OR 图所做的研究有助于人工智能中的搜索技术的进一步深入研究和发展.

参考文献

1 王士同等编著.模糊数学在人工智能中的应用.北京：机械工业出版社，1991.

2 王士同.模 S 下的 AND/OR 图的启发式搜索算法 NAO*.计算机学报，1991；**14**(1)：14—22.

3 王士同.随机产生式系统的启发式图搜索算法 RA* 及 A* 的推广.计算机学报，1988；**11**(5)：294—299.

4 王士同.启发式图搜索算法 RA* 之改进算法 IRA* 及 IRA′.计算机学报，1991；**14**(3)：192—198.

5 王士同.双向启发式图搜索算法 BRA* 之研究.计算机学报，1993；**16**(1).

6 Nilsson N J. Principles of artificial intelligence. Tioga Publishing Co., 1980.

7 Pearl J. Heuristics. Intelligent search strategies for computer problem solving. Addison—Wesley Press, 1984.

8 王士同.传播式启发式图搜索算法 PRA 及 PRA*.软件学报，1992；**3**(1)：49—54.

9 张文修.模糊数学基础.西安：西安交通大学出版社，1984.

10 Mero L. Some remarks on heuristic search algorithms. Proc. of IJCAI—81, Canada, 1981.

BACKWARD HEURISTIC SEARCH ALGORITHM BHAO* FOR GENERAL AND/OR GRAPH

Wang Shitong

(*Department of Computer Science, Zhenjiang Shipbuilding Institute, Zhenjiang* 212003)

Abstract In this paper, general AND/OR graphs of a new type are defined on triangle norm. The backward heuristic search algorithm BHAO* for general AND/OR graphs is presented, in terms of newly—defined heuristic function h(n,x) and the principle that every subtree of the optimum solution tree is also optimum solution one. The admissibility of algorithm BHAO* is proved. Based on two newly — defined monotone restrictions for heuristic function h(n,x), the characteristic of monotone restrictions for algorithm BHAO* is also investigated, and the comparison between two BHAO* algorithms is also discussed.

Key words General AND/OR graph, heuristic search, heuristic function, monotone restriction, algorithm.