2D−7

マルチプロセッサシステムにおける多段先行評価制御手順

石井吉彦　安江俊明　山名早人　村岡洋一
e-mail:harray@muraoka.info.waseda.ac.jp
(早稲田大学　理工学研究科)

1．はじめに

本稿では，マルチプロセッサシステムにおける，タスク一段の先行評価の制御（一段先行評価制御）と，タスク多段に渡る先行評価の制御（多段先行評価制御）との違いについて述べる．

我々は，一段先行評価制御，及び，多段先行評価制御を具体的なマルチプロセッサシステム（並列処理システムー晴ー[1]）に沿って提案してきた[2][3]．

本稿では，一段先行評価制御，及び，多段先行評価制御を時相論理で表現し一般化する．その後，この時相論理を用いて，制御の違いを推論する．また，この推論によって，我々が提案してきた具体的なマルチプロセッサシステムに沿った一段先行評価制御，及び，多段先行評価制御の正当性を述べる．

2．一段先行評価制御，多段先行評価制御

要素プロセッサを$p \in P$（P：要素プロセッサ全体），タスクを$t \in T$（T：タスク全体）で表す．Pの各要素プロセッサpに，それぞれTの１つのタスクtが割り当てられているとき，この対応をSで表し，PからTへの写像（$S:P \to T$）といい，$S(P)=\{S(p):p \in P\}$，$S^{-1}(T)=\{p:S(p) \in T\}$で表す．

本稿では，マクロブロックをタスクとしたマルチプロセッサシステムを想定しているため，$S(p)=t$は全射（上への写像）であり，単射ではない．

また，要素プロセッサpに割り当てられているタスクの時系列が，$\{t_1,t_2,\cdots,t_n,\cdots:t_n \in T\}$で表されるとき，nを$t_n$の世代といい，世代nにおいて，要素プロセッサpにタスクt_nが割り当てられていることを，$S_n(p)=t_n$で表す．従って，

$$\forall n\ \forall t_n \in T\ \exists p \in P\{S_n(p)=t_n\}$$

が成り立つ．

また，要素プロセッサpに割り当てられているタスクの処理状態には，先行評価と非先行評価の２つがあり，

命題E(p)：「要素プロセッサpはタスクを先行評価で処理」
命題N(p)：「要素プロセッサpはタスクを非先行評価で処理」

で表す．このとき，

$$E(p)'=N(p)$$

が成り立つ．

また，世代nのタスクt_nから世代mのタスクt_mへの制御依存があるとき，この制御依存を$t_n \delta t_m$で表す．なお，マルチプロセッサシステムにおいて（非先行評価では）制御依存$t_n \delta t_m$の世代n,mは計算順序を保証するため，n<mを満たさなければならない．従って，mが最小になるようにスケジューリングするとm=n+1となる．

本稿では，便宜上，

$$\forall n\ \exists ti_n,tj_{n+1} \in T\{ti_n \delta tj_{n+1}\}\ \exists pi,pj \in P\{Si_n(pi)=t_n,S_{n+1}(pj)=t_{n+1}\}$$

を$pi_n \Delta pj_{n+1}$で表す．なお，「mが最小になるようにスケジューリングする」とは，

$$pi_n \Delta pj_m \rightarrow pi_n \Delta pj_{n+1} \qquad (n<m) \qquad \text{式(1)}$$

のことである．

以上より，一段先行評価制御を表す命題SE(pi,pj)は，

$$pi_n \Delta pj_{n+1} \wedge N(pi) \wedge XN(pj) \wedge pi \neq pj \rightarrow pi_n \Delta pj_n \wedge N(pi) \wedge E(pj) \qquad \text{式(2)}$$

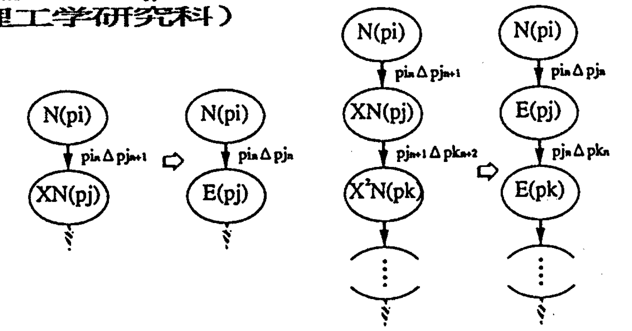
で表される．（図１(a)参照）

一方，多段先行評価制御を表す命題ME(pi,pj,pk,…)は，

$$pi_n \Delta pj_{n+1} \Delta pk_{n+2} \Delta \cdots \wedge N(pi) \wedge XN(pj) \wedge X^2N(pk) \wedge \cdots \wedge \forall \alpha,\beta \in i,j,k,\cdots\{p\alpha \neq p\beta\} \rightarrow pi_n \Delta pj_n \Delta pk_n \Delta \cdots \wedge N(pi) \wedge E(pj) \wedge E(pk) \wedge \cdots \qquad \text{式(3)}$$

で表される．（図１(b)参照）

A Control Scheme of Multistage Eager Evaluation for Multiprocesser System
Yoshihiko ISHII,Toshiaki YASUE,Hayato YAMANA, Yoichi MURAOKA
Graduate School of Science and Engineering,Waseda University



(a)一段先行評価制御　(b)多段先行評価制御
図１　先行評価制御

本稿で明らかにしたいのは，一段先行評価制御SE(pi,pj)と多段先行評価制御ME(pi,pj,pk,…)との制御の違いである．仮に，

$$ME(pi,pj,pk,\cdots)=SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots$$

が成り立てば，制御の違いは存在しない．

以下，２．１では，一段先行評価制御SE(pi,pj)を恒真命題としたときの制御の違いを述べ，２．２では，逆に，多段先行評価制御ME(pi,pj,pk,…)を恒真命題としたときの制御の違いを述べる．２．３では，２．１の制御と２．２の制御を比較する．

２．１　恒真命題，一段先行評価制御SE(pi,pj)

いま，一段先行評価制御SE(pi,pj)を恒真命題としたときの，制御の違いを表す命題をQ(pi,pj,pk,…)とすると，

$$\forall pi,pj,pk,\cdots \in P\{SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}\ \exists Q(pi,pj,pk,\cdots)\{ME(pi,pj,pk,\cdots) \wedge Q(pi,pj,pk,\cdots)=SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}$$

付録(1)より，

$$\Rightarrow \forall pi,pj,pk,\cdots \in P\{SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}\ \exists Q(pj)\{E(pj) \wedge Q(pj)=N(pj)\} \wedge \exists Q(pk)\{E(pk) \wedge Q(pk)=N(pk)\} \wedge \cdots$$

$$\Rightarrow \forall pi,pj,pk,\cdots \in P\{SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}\ N(pj) \wedge N(pk) \wedge \cdots \wedge \exists Q(pj)\{E(pj) \wedge Q(pj)=N(pj)\} \wedge \exists Q(pk)\{E(pk) \wedge Q(pk)=N(pk)\} \wedge \cdots$$

$$\Leftrightarrow \forall pi,pj,pk,\cdots \in P\{SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}\ E(pj)' \wedge E(pk)' \wedge \cdots \wedge \exists Q(pj)\{E(pj) \wedge Q(pj)=N(pj)\} \wedge \exists Q(pk)\{E(pk) \wedge Q(pk)=N(pk)\} \wedge \cdots$$

$$\Leftrightarrow \forall pi,pj,pk,\cdots \in P\{SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots\}\ \exists Q(pj)\{E(pj) \wedge Q(pj) \rightarrow N(pj)\} \wedge \exists Q(pk)\{E(pk) \wedge Q(pk) \rightarrow N(pk)\} \wedge \cdots$$

が成り立つ．

この制御の違いを表す命題$Q(p)\{E(p) \wedge Q(p) \rightarrow N(p)\}$の存在は，「多段先行評価制御ME(pi,pj,pk,…)において，本来，タスクを先行評価で処理しているpj,pk,…を，非先行評価で処理しているとみなせば，式(2)で示した一段先行評価制御SE(pi,pj),SE(pj,pk),…がそのまま適用できること」を意味している．（図２(a)参照）

２．２　恒真命題，多段先行評価制御ME(pi,pj,pk,…)

いま，多段先行評価制御ME(pi,pj,pk,…)を恒真命題としたときの，制御の違いを表す命題をR(pi,pj,pk,…)とすると，

$$\forall pi,pj,pk,\cdots \in P\{ME(pi,pj,pk,\cdots)\}\ \exists R(pi,pj,pk,\cdots)\{ME(pi,pj,pk,\cdots)=SE(pi,pj) \wedge SE(pj,pk) \wedge \cdots \wedge R(pi,pj,pk,\cdots)\}$$

付録(2)より，

$$\Rightarrow \forall pi,pj,pk,\cdots \in P\{ME(pi,pj,pk,\cdots)\}\ \exists R(pj)\{E(pj)=N(pj) \wedge R(pj)\} \wedge \exists R(pk)\{E(pk)=N(pk) \wedge R(pk)\} \wedge \cdots$$

$$\Rightarrow \forall pi,pj,pk,\cdots \in P\{ME(pi,pj,pk,\cdots)\}\ E(pj) \wedge E(pk) \wedge \cdots \wedge \exists R(pj)\{E(pj)=N(pj) \wedge R(pj)\} \wedge \exists R(pk)\{E(pk)=N(pk) \wedge R(pk)\} \wedge \cdots$$

$$\Leftrightarrow \forall pi,pj,pk,\cdots \in P\{ME(pi,pj,pk,\cdots)\}\ N(pj)' \wedge N(pk)' \wedge \cdots \wedge \exists R(pj)\{E(pj)=N(pj) \wedge R(pj)\} \wedge \exists R(pk)\{E(pk)=N(pk) \wedge R(pk)\} \wedge \cdots$$

$$\Leftrightarrow \forall pi,pj,pk,\cdots \in P\{ME(pi,pj,pk,\cdots)\}\ \exists R(pj)\{E(pj) \leftarrow N(pj) \wedge R(pj)\} \wedge \exists R(pk)\{E(pk) \leftarrow N(pk) \wedge R(pk)\} \wedge \cdots$$

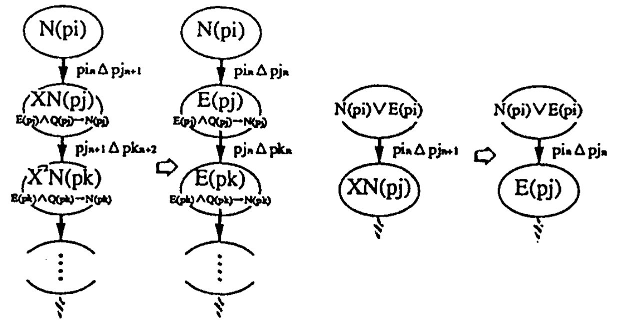
が成り立つ．

この制御の違いを表す命題$R(p)\{E(p) \leftarrow N(p) \wedge R(p)\}$の存在は，「一

段先行評価制御SE(pj,pk),SE(pk,pl),…において，本来，タスクを非先行評価で処理しているpj,pk,…を，先行評価で処理しているとみなせば，式(3)で示した多段先行評価制御ME(pi,pj,pk,…)がそのまま適用できること」を意味している．しかし，一段先行評価制御SE(pi,pj)では式(2)が成り立つ．従って，式(2)で示した一段先行評価制御を表す命題SE(pi,pj)は，

$$pi_n\Delta pj_{n+1}\wedge\{N(pi)\vee E(pi)\}\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge\{N(pi)\vee E(pi)\}\wedge E(pj) \qquad 式(4)$$

となる．(図2(b)参照)



(a)恒真命題，一段先行評価制御　(b)恒真命題，多段先行評価制御
図2　制御の違い

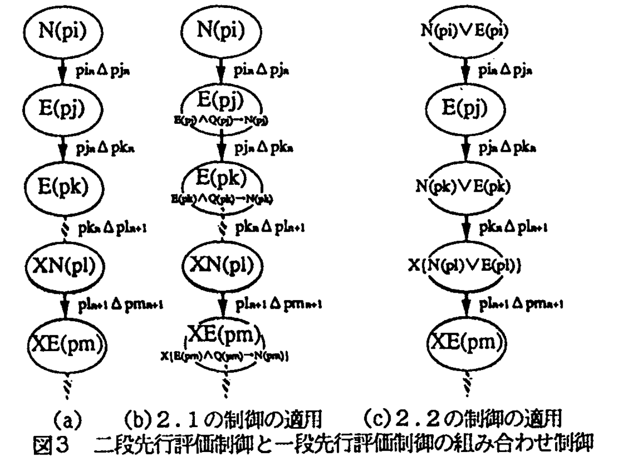
2.3　2.1の制御と2.2の制御との比較

二段先行評価制御と一段先行評価制御を組み合わせた制御を図3(a)に示す．

図3(a)に2.1の制御を適用すると図3(b)となり，「二段先行評価制御でE(pk)∧Q(pk)→N(pk)が成り立つ(pkが先行評価から非先行評価になる)とき，制御依存$pk_n\Delta pl_{n+1}$によりpkからplに制御依存情報が放送され，次世代の一段先行評価制御に制御が移ること」を示している．E(pk)∧Q(pk)→N(pk)が成り立つかどうかといった制御が追加されるため，2.1の制御にはオーバーヘッドがある．しかし，pkが先行評価から非先行評価にならないときには，pkからplに制御依存情報が放送されないため，plの計算機資源を消費しない．そのため，2.1の制御には計算機資源の無駄がない．

一方，図3(a)に2.2の制御を適用すると，式(4)より，図3(c)となり，「二段先行評価制御でN(pk)∨E(pk)が成り立つとき，制御依存$pk_n\Delta pl_{n+1}$によりpkからplに制御依存情報が放送され，次世代の一段先行評価制御に制御が移ること」を示している．N(p)∨E(p)は恒真命題なので，N(pk)∨E(pk)といった制御は削除されるため，2.2の制御にはオーバーヘッドがない．しかし，pkが先行評価から非先行評価にならない(次世代の一段先行評価制御に制御が移ることはない)ときにも，pkからplに制御依存情報が放送され，plの計算機資源を消費する．そのため，2.2の制御には計算機資源の無駄がある．

我々が開発している並列処理システム－晴－は，制御による計算機資源の無駄が許されるほど計算機資源が多くないマルチプロセッサシステムである．そのため，－晴－では，2.1の制御を用いて，一段先行評価制御，多段先行評価制御を具現化している．しかし，制御による計算機資源の無駄が許されるほど計算機資源が多いマルチプロセッサシステム(超並列処理システム)では，制御のオーバーヘッドを考慮して，2.2の制御を用いるべきである．



(a)　(b)2.1の制御の適用　(c)2.2の制御の適用
図3　二段先行評価制御と一段先行評価制御の組み合わせ制御

3．おわりに

本稿では，一段先行評価制御，及び，多段先行評価制御を時相論理で表現し一般化した．その後，この時相論理を用いて，制御の違いは，「先行評価を非先行評価とみなすこと」，または，「非先行評価を先行評価とみなすこと」と推論した．前者の制御の違いの，先行評価制御の組み合わせへの適用では，制御にオーバーヘッドがあるが，計算機資源の無駄がない．一方，後者の制御の違いの適用では，制御にオーバーヘッドがないが，計算機資源の無駄がある．我々が開発している並列処理システム－晴－では，制御による計算機資源の無駄が許されないので，－晴－の一段先行評価制御，多段先行評価制御は前者の制御の違いを用いて具現化している．

謝辞

本研究の遂行にあたり，数々の御助言を頂いた本研究室の石崎一明氏，及び，萩本猛氏をはじめとする－晴－グループの皆さんに感謝致します．

参考文献

[1]H.Yamana,et al.:"Parallel Processing System -Harray-",Computing System in Engineering Vol.1,No.1,pp.111-130(1990)
[2]石崎他："並列処理システム－晴－の要素プロセッサ制御方式"，第42回情処全大，4H-2(1991)
[3]石井他："並列処理システム－晴－における要素プロセッサ制御手順"，第43回情処全大，4Q-1(1991)

付録(1) $ME(pi,pj,pk,\cdots)\wedge Q(pi,pj,pk,\cdots) = SE(pi,pj)\wedge SE(pj,pk)\wedge\cdots$

$$\Leftrightarrow \{pi_n\Delta pj_{n+1}\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge XN(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\}\wedge Q(pj,pk,\cdots) = \{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}\wedge\{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots$$

$$\Leftrightarrow \underline{\{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}}\wedge\{pi_n\Delta pj_n\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\}\wedge Q(pi,pj,pk,\cdots) = \underline{\{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}}\wedge\{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots$$

$$\Leftrightarrow \{pi_n\Delta pj_n\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\}\wedge Q(pi,pj,pk,\cdots) = \{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots$$

式(1)より，

$$\Leftrightarrow \{pi_n\Delta pj_n\Delta pk_{n+1}\Delta\cdots\wedge N(pi)\wedge \mathit{E(pj)}\wedge XN(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge\mathit{E(pj)}\wedge E(pk)\wedge\cdots\}\wedge Q(pi,pj,pk,\cdots) = \{pj_n\Delta pk_{n+1}\wedge\mathit{N(pj)}\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge\mathit{N(pj)}\wedge E(pk)\}\wedge\cdots$$

$$\Rightarrow \{E(pj)\wedge E(pk)\wedge\cdots\}\wedge Q(pi,pj,pk,\cdots) = \{N(pj)\wedge N(pk)\wedge\cdots\}$$

$$\Leftrightarrow \{E(pj)\wedge Q(pj)=N(pj)\}\wedge\{E(pk)\wedge Q(pk)=N(pk)\}\wedge\cdots$$

∵Q(pi,pj,pk,…)はpj,pk,…に対して独立

付録(2) $ME(pi,pj,pk,\cdots) = SE(pi,pj)\wedge SE(pj,pk)\wedge\cdots\wedge R(pi,pj,pk,\cdots)$

$$\Leftrightarrow \{pi_n\Delta pj_{n+1}\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge XN(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\} = \{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}\wedge\{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots\wedge R(pj,pk,\cdots)$$

$$\Leftrightarrow \underline{\{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}}\wedge\{pi_n\Delta pj_n\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\} = \underline{\{pi_n\Delta pj_{n+1}\wedge N(pi)\wedge XN(pj)\wedge pi\neq pj\rightarrow pi_n\Delta pj_n\wedge N(pi)\wedge E(pj)\}}\wedge\{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots\wedge R(pi,pj,pk,\cdots)$$

$$\Leftrightarrow \{pi_n\Delta pj_n\Delta pk_{n+2}\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge X^2N(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge E(pj)\wedge E(pk)\wedge\cdots\} = \{pj_n\Delta pk_{n+1}\wedge N(pj)\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge N(pj)\wedge E(pk)\}\wedge\cdots\wedge R(pi,pj,pk,\cdots)$$

式(1)より，

$$\Leftrightarrow \{pi_n\Delta pj_n\Delta pk_{n+1}\Delta\cdots\wedge N(pi)\wedge\mathit{E(pj)}\wedge XN(pk)\wedge\cdots\wedge\forall\alpha,\beta\in i,j,k,\cdots\{p\alpha\neq p\beta\}\rightarrow pi_n\Delta pj_n\Delta pk_n\Delta\cdots\wedge N(pi)\wedge\mathit{E(pj)}\wedge E(pk)\wedge\cdots\} = \{pj_n\Delta pk_{n+1}\wedge\mathit{N(pj)}\wedge XN(pk)\wedge pj\neq pk\rightarrow pj_n\Delta pk_n\wedge\mathit{N(pj)}\wedge E(pk)\}\wedge\cdots\wedge R(pi,pj,pk,\cdots)$$

$$\Rightarrow \{E(pj)\wedge E(pk)\wedge\cdots\} = \{N(pj)\wedge N(pk)\wedge\cdots\}\wedge R(pi,pj,pk,\cdots)$$

$$\Leftrightarrow \{E(pj)=N(pj)\wedge R(pj)\}\wedge\{E(pk)=N(pk)\wedge R(pk)\}\wedge\cdots$$

∵R(pi,pj,pk,…)はpj,pk,…に対して独立