数理解析研究所講究録
第 1554 巻 2007 年 202-209

確率的枝重みを持つグラフ上での最小全域木コストの近似見積り法に関する考察

安藤 映 (Ei Ando) 小野 廣隆 (Hirotaka Ono)
定兼 邦彦 (Kunihiko Sadakane) 山下 雅史 (Masafumi Yamashita)

Department of Computer Science and Communication Engineering,
Graduate School of Information Science and Electrical Engineering,
Kyushu University

概要

本稿では枝重みが互いに独立な確率変数として与えられる場合の最小全域木問題に関して，その枝重み合計の分布を近似的に求める手法について考察する．アルゴリズムの設計に当たって，matrix-tree 定理を一般化した定理を証明し，全ての全域木の重み分布についてその最小値のとる分布を行列式演算の一般化として表現する．

本稿では，最小全域木の重み合計が高々 x である確率 $F(x)$ に着目して，与えられたパラメータ a について x が $F^{-1}(1-a)$ よりも小さいところで分布関数の上限を与えるような正規分布関数 $\tilde{F}(x)$ を得るような近似アルゴリズムを目標とする．しかし，正規分布関数 $\tilde{F}(x)$ を多項式時間で得るようなアルゴリズムとするにはもう一段階の工夫を加える必要がある．

1 はじめに

無向グラフの最小全域木を求める問題はグラフ上の問題としてもっとも古い部類に属する問題で，枝重みが確定的な値として与えられる場合には高速な解法が提案されている．現在最速のアルゴリズムは Chazelle[1] によるものが知られている．

無向グラフ G において枝重みがある分布に従う確率変数として与えられる場合を考える．このとき，最小全域木の枝重み合計もある確率変数となるが，枝重みが正規分布に従うと仮定しても G が閉路を含む場合には，最小全域木の重み分布は正規分布ではなくなるため，一般にその分布を正確に求めることは難しい．

Kulkarni[2] は枝重みが指数分布に従い，互いに独立な確率変数と仮定し，その最小全域木の分布および任意の枝が最小全域木に含まれる確率を得る方法を提案している．

本稿では分布関数の上限を高速に近似する手法について考察する．全域木の数を計算する項式の一般化に基づいて，matrix-tree 定理を一般化した定理を証明し，全ての全域木の重み分布を行列式演算の一般化として表現する．

本稿では，最小全域木の重み合計が高々 x である確率 $F(x)$ に着目して，与えられたパラメータ a について x が $F^{-1}(1-a)$ よりも大きいところで分布関数の上限を与えるような正規分布関数 $\tilde{F}(x)$ を得るような近似アルゴリズムを目標とする．しかし，正規分布関数 $\tilde{F}(x)$ を多項式時間で得るようなアルゴリズムとするにはもう一段階の工夫を加える必要がある．

2 準備

平均 μ，分散 σ^2 の正規分布 $N(\mu,\sigma^2)$ の確率密度関数 $f(x)$ は次式で与えられる．

$$f(x)=\frac{1}{2\pi\sigma}\exp\left(-\frac{x^2}{2\sigma}\right)$$

二つの確率変数 X_1, X_2 がそれぞれ正規分布 $N(\mu_1,\sigma_1^2)$, $N(\mu_2,\sigma_2^2)$ に従う場合は，その和 X_1+X_2 は正規分布 $N(\mu_1+\mu_2,\sigma_1^2+\sigma_2^2)$ に従うことが知られている．このためグラフ G が木である場合には枝重みの平均と分散の和をとることで最小全域木の厳密な分布を計算できる．

グラフ G が単一の閉路からなる場合，グラフ G から任意のひとつの枝を取り除くことで全域木が得られるため，次のような計算で最小全域木の分布関数 $F(x)$ は，次の式で上限 $F_U(x)$ を得ることができる．

$$F_U(x)=1-\prod_{1\le i\le |E|}(1-F_i(x))$$

ただし $F_i(x)$ は枝 $e_i \in E\,(1\le i\le |E|)$ 以外の枝重みの合計の分布関数とする．各枝重みが正規分布に従う確率変数である場合でも $F_U(x)$ は正規分布関数ではない．

3 高速に分布関数の上限を近似する手法

関数 $F(x)$ を最小全域木の重み分布関数とする．関数 $F(x)$ の上限を求めることは全ての全域木を列挙すれば可能であるが，全域木の数は最大で $|V|^{|V|-2}$ 個あり，効率的なアルゴリズムとは言えない．本節ではより効率的に分布関数の上限を求めることを考える．

3.1 matrix-tree 定理の一般化

全ての全域木の数であれば行列式の計算によって得ることが知られている（matrix-tree 定理）．まず，matrix-tree 定理について説明し，次に matrix-tree 定理の一般化から分布関数の上限を近似する手法を得ることを考える．

定理 1 (matrix-tree 定理)

グラフ G から任意の 1 頂点 v を取り除いたグラフ $\tilde{G}_v$ の隣接行列 A と次の式によって定義される $(n-1)\times(n-1)$ 行列 D を考える．ただし d_i は頂点 i の $\tilde{G}_v$ 中での次数，d_{ij} は行列 D の i 行 j 列目の成分である．

$$d_{ij}=\begin{cases} d_i & i=j \text{ の場合}\\ 0 & i\neq j \text{ の場合}\end{cases}$$

グラフ G 中の全域木の集合を $\mathcal{T}$ とすると $|\mathcal{T}|$ は次の式で表され，取り除かれる頂点 v には依存しない．

$$|\mathcal{T}| = \det(D - A)$$

まず，グラフ G から任意の 1 頂点 v を取り除いたグラフ $\tilde{G}_v$ のについて E の部分集合 E'_v に対して次の式で定義される $(n-1)\times(n-1)$ 行列 $H^{E'}$ を考える．ただし $h_{ij}^{E'v}$ は $H^{E'}$ の i 行 j 列目の成分であるとする．

$$\begin{cases} h_{ij}^{E'} = \sum_{k=1}^{n} \delta_{ik} & i = j \text{ の場合} \\ h_{ij}^{E'} = -\delta_{ij} & i \neq j \text{ の場合} \end{cases}$$

この式で $\{i,j\} \notin E'_v$ の場合のみ $\delta_{ij}^{E'v} = 0$ として，$\det(H^E)$ の計算を考える．

定理 2 (matrix-tree 定理の一般化)

$\det(H^{E'v})$ を展開して積和の形に整理すると次式で表される．

$$\sum_{T\in\mathcal{T}} \prod_{e\in T} \delta_e^{E'v}$$

ただし $\mathcal{T}$ は G の全域木の集合である．

証明 (概略)

行列式演算の性質により，積和形に展開した場合はどの項も各行各列からひとつ要素を因数として持つ．また，同じ行に属する二つの要素 a_{ij}, a_{ik} の積を因数とする項は存在せず，同じ列に属する二つの要素 a_{ji}, a_{ki} の積を因数とする項も存在しない．従って，$n-1$ 個の頂点から K_n の枝集合 $V\times V$ への写像 $l : V\setminus\{v\} \mapsto V^2$ を考えると，その l 中でも特に δ_{ij}^{E} を因数として持つ項と頂点 i を枝 $\{i,j\}\in E$ へ写すような写像 l_E と対応付けることができる．

以下，写像 l_E で写される先の枝集合 $E_l \subseteq E$ に着目する．いま，E_l が閉路 $C \subseteq E$ を含むと仮定する．すると，行列 H^E 中での頂点の順番の入れ換えは $\det(H^E)$ の値を変化させないため，一般性を失うことなく閉路 C の内部頂点は $v_1, v_2, ..., v_{|C|}$ であると考えることができる．

ここで，$\det(H^E)$ の閉路 $|C|$ に関わる部分だけを強調すると，概観は次式 (1) のような形となる．

$$\begin{vmatrix} \Delta_1^E & -\delta_{1,2}^E & 0 & \cdots & \cdots & 0 & -\delta_{1,|C|}^E \\ -\delta_{2,1}^E & \Delta_2^E & & & & & 0 \\ 0 & & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & & 0 \\ 0 & & & & & \Delta_{|C|-1}^E & -\delta_{|C|-1,|C|}^E \\ -\delta_{|C|,1}^E & 0 & \cdots & \cdots & 0 & -\delta_{|C|-1,|C|}^E & \Delta_{|C|}^E \end{vmatrix} \quad (1)$$

ただし，$\Delta_i^E = \sum_{j=1}^{n} \delta_{i,j}^E$ である．

ここで，$\det(H^E)$ の対角成分の積の中から，閉路 C の枝に対応する δ^E_e を全て因数に持つ項だけ抜き出すことを考える．共通の因数を小行列式を用いて括り出してそれらの項をまとめると次式 (2) のように書ける．

$$\delta^E_{|C|,1}\prod_{k=1}^{|C|}\delta^E_{k-1,k}\begin{vmatrix} \Delta^E_{|C|+1} & -\delta^E_{|C|+1,|C|} & \cdots & & -\delta^E_{|C+1|,n-1} \\ -\delta^E_{|C|+2,1} & \Delta^E_{|C|+2} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \Delta^E_{n-2} & -\delta^E_{n-2,n-1} \\ -\delta^E_{n-1,1} & \cdots & \cdots & -\delta^E_{n-1,n-2} & \Delta^E_{n-1} \end{vmatrix} \tag{2}$$

以下，対角成分の積以外で，$V\setminus\{v_n\}$ から閉路 C への写像に対応する項が存在し，対角成分に存在する項と符号が逆であることを示す．閉路には枝 $\{1,2\}$ が含まれるので，行列式 (1) の 1 行 2 列目の要素 $\delta^E_{1,2}$ に着目し，これを因数として持つ部分を小行列式として書くと，次式 (3) の通りとなる．この行列式の係数は負号が相殺して正である．

$$\delta^E_{1,2}\begin{vmatrix} -\delta^E_{2,1} & -\delta^E_{2,3} & 0 & \cdots & \cdots & 0 & -\delta^E_{1,|C|} \\ -\delta^E_{3,1} & \Delta^E_3 & & & & & 0 \\ 0 & & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & & 0 \\ 0 & & & & & \Delta^E_{|C|-1} & -\delta^E_{|C|-1,|C|} \\ -\delta^E_{|C|,1} & 0 & \cdots & \cdots & 0 & -\delta^E_{|C|-1,|C|} & \Delta^E_{|C|} \end{vmatrix} \tag{3}$$

同様に $\delta^E_{2,3}$，$\delta^E_{3.4}\ldots$ を因数に持つ部分を抜き出すと，$\prod_{k=1}^{|C|}\delta^E_{k-1,k}$ の因数を持つ項は小行列式を使って次式 (4) によって表される．

$$\left|\begin{array}{c|ccccc} -\delta^E_{|C|,1} & -\delta^E_{|C|,|C|+1} & \cdots & \cdots & \cdots & -\delta^E_{|C|,n-1} \\ \hline -\delta^E_{|C|+1,1} & \Delta^E_{|C|+1} & -\delta^E_{|C|+1,|C|+2} & \cdots & & -\delta^E_{|C|+1,n-1} \\ \vdots & -\delta^E_{|C|+2,|C|+1} & \Delta_{|C|+2} & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \Delta^E_{n-2} & -\delta^E_{n-2,n-1} \\ -\delta^E_{n-1,1} & -\delta^E_{n-1,|C|+1} & \cdots & \cdots & -\delta^E_{n-1,n-2} & \Delta^E_{n-1} \end{array}\right| \times\prod_{k=1}^{|C|}\delta^E_{k-1,k} \tag{4}$$

ここで，1 行 1 列目の $\delta^E_{|C|-1,1}$ を因数として持つ部分を抜き出すと，負号が相殺されないため小行列式の係数は -1 であることがわかる．従って，式 (2) と式 (4) を比較すると符号だけが異なり，残りは全て同じ式であるためにこれら二つの項は互いに相殺する．

以上により，$\det(H^E)$ を積和形に展開して得られる式中のひとつの項と対応する写像 $l : V \setminus \{v_n\} \mapsto E_l$ について E_l が閉路を含む場合は必ず対角成分と，対角成分以外にひとつずつ符号の違う成分が存在して互いに相殺することが示せた．

従って，$V \setminus \{v_n\}$ から閉路のある E_l への写像 l に対応する項は全て互いに相殺するため，閉路の無い E_l すなわち全域木に対応する項だけが残る．■

ある枝 $e \in E$ に関して $\delta_e^E = 1$ とすると定理 2 から全域木の数が得られるため定理 2 は matrix-tree 定理の一般化と考えることができる．次節以降では，確率分布を行列の要素として持つ行列を考え，行列式演算の一般化を適用することで最小全域木の分布を表現することを説明する．

3.2 行列式演算の一般化

本節では，行列式演算を行列 M，加算記号 $\boxplus$，乗算記号 $\boxtimes$ の 3 つに対する一般化 $DET(M, \boxplus, \boxtimes)$ を考える．

定義 (行列式演算の一般化)

行列式演算の一般化 $\mathrm{DET}(M, \boxplus, \boxtimes)$ を次のように再帰的に定義する．ただし，行列 M は $m \times m$ 行列と仮定し，行列 M の i 行 j 列目の要素を m_{ij} と書く．減算"$\boxminus$" は加算記号 $\boxplus$ の逆演算，Σ は合計演算の一般化で $\Sigma_{i=1}^n A_i = A_1 \boxplus A_2 \boxplus \cdots \boxplus A_n$ である．

$$\mathrm{DET}(M, \boxplus, \boxtimes) = \Sigma_{i=1}^m (\boxminus 1)^{i-1} m_{1i} \boxtimes \mathrm{DET}(M_i, \boxplus, \boxtimes)$$

小行列 M_i は行列 M から i 行と i 列を削除して得られる行列である．

例えば，行列 M の行列式は DET を用いて $\mathrm{DET}(M, +, \times)$ と書くことができる．

3.3 最小全域木の重み分布の定式化

ここから，確率分布の計算をする演算子として $\boxtimes$ と $\boxplus$ を定義する．

定義 演算子 $\boxtimes$(二つの確率変数の和の分布)

二つの独立な確率変数 X_1，X_2 の従う確率分布 D_1，D_2 について，$D_1 \boxtimes D_2$ は二つの確率変数の和 $X_1 + X_2$ の従う分布である．特に，二つの確率分布の密度関数が $f_1(x)$，$f_2(x)$ であるとき，$D_1 \boxtimes D_2$ の密度関数 $f(x)$ は，畳み込み積分を用いて次式で表される．

$$f(x) = \int_{-\infty}^{\infty} f_1(x - \tau) f_2(\tau) d\tau$$

定義 演算子 $\boxplus$(二つの確率変数の最小値の分布)

二つの独立な確率変数 X_1，X_2 の従う確率分布 D_1，D_2 について，$D_1 \boxplus D_2$ は二つの確率変数の最小値 $\min\{X_1, X_2\}$ の従う分布である．特に，二つの確率分布の分布関数が $F_1(x)$，$F_2(x)$ であるとき，$D_1 \boxplus D_2$ の分布関数 $F(x)$ は次式で表される．

$$F(x) = 1 - (1 - F_1(x))(1 - F_2(x))$$

この関係は，補分布関数 $F_1^C(x) = 1 - F_1(x)$，$F_2^c(x) = 1 - F_2(x)$，$F^C(x) = 1 - F(x)$ を用いることによって次のように簡潔に書ける．

$$F^C(x) = F_1^C(x) F_2^C(x)$$

すなわち $\min\{X_1, X_2\}$ の補分布関数 $F^C(x)$ は補分布関数 $F_1^C(x)$, $F_2^C(x)$ の積である.

定理2及びDET, ⊠, ⊞ を用いると最小全域木の枝重み和の分布を次のように matrix-tree 定理と同様の形式で書き表すことができる.

定理 3(最小全域木の重み分布)

枝 $\{i,j\}$ の枝重みが従う確率分布を $\delta_{i,j}$ とおき, $\Delta_{i,j} = \Sigma_{j=1}^{n}\delta_{i,j}$ と定義する. ただし, $\{i,j\} \notin E$ の場合, 枝重みはある確率分布 Z に従う確率変数とし, Z の確率分布関数 $F_Z(x)$ はある十分大きな実数値 x_Z に対して, $x < x_Z$ のとき $F_z(x) = 0$ とする. 次の式で表される確率分布の $(n-1)\times(n-1)$ 行列 P を考える. ただし, p_{ij} は行列 P の i 行 j 列目の要素とする.

$$p_{ij} = \begin{cases} \Pi_{j=1}^{n}\delta_{i,j} & i = j \text{ の場合} \\ \boxminus\delta_{i,j} & i \neq j \text{ の場合} \end{cases}$$

ただし, ⊟ は ⊞ 演算の逆演算, Π は積の一般化で $\Pi_{i=1}^{n} A_i = A_1 \boxplus A_2 \boxplus \cdots \boxplus A_n$ である. このとき, 最小全域木の重み分布は次式で表される.

$$\mathrm{DET}(P, \boxplus, \boxtimes)$$

この定理においては, 演算 ⊞ を加算の代わり, ⊠ を乗算の代わりに使うことに気を付ける.

証明

定理2より, DET を展開して得られる各項は次式の形となる.

$$\Pi_{T\in\mathcal{T}}\Sigma_{e\in T}\delta_e$$

これは全ての全域木を列挙して, その重み分布の最小値を取ることに対応する. ■

定理3から, 行列式計算の一般化である DET を計算できれば, 最小全域木の分布を求めることができるとわかる. なお, 定理3は枝重みが正規分布の場合に限らず, 一般の分布で通用する. ただし, 一般に min 演算には補分布関数 (complementary distributio function) の積を取ることが必要であり, その計算には因数の補分布関数の数と同じだけの計算時間がかかる. 従って, 定理3による計算だけでは少なくとも全域木の数と同じだけの計算時間がかかるため効率的な方法とは言えない.

3.4 最小値演算の近似

グラフ G の枝重みが互いに独立で正規分布に従う確率変数として与えられている場合, 定理3に沿って最小全域木の枝重みを計算するには補分布関数の積を計算する際に全因数をリストとして保持しておく必要があり, 効率的な手法とはまだいえない. 以下では, 演算 ⊞ を近似することによって, 次の方針を満たす正規分布関数を求め, 正規補分布関数の積を抑えること考える.

方針

グラフ G の枝重みが互いに独立で正規分布に従う確率変数として与えられている場合, ある定数 a $(0 \leq a \leq 1)$ に対して, 次式を満たす正規分布関数 $\tilde{F}(x)$ を求める. ただ

し $F(x)$ はグラフ G の全域木の重みの分布関数.

$$x \leq F^{-1}(1-a) \Rightarrow F(x) \leq \tilde{F}(x)$$

方針で求められる関数の値は，直観的に言えば確率 a で最小全域木の重みよりも小さな値を見積もることに対応する．

二つの正規分布関数 $F_1(x)$，$F_2(x)$ の最小値演算結果をひとつの $\tilde{F}(x)$ で置き換えることを考える．この手法は [3] と同じ手法であるので簡潔に説明する．

定理 4(正規分布関数による ⊞ 演算の近似)

正規分布 $N(\mu_1, \sigma_1^2)$ の分布関数を $F_1(x)$，$N(\mu_2, \sigma_2^2)$ の分布関数を $F_2(x)$ とおく．一般性を失うことなく $\sigma_1 \geq \sigma_2$ とする．$N(\mu_1, \sigma_1^2) \boxplus N(\mu_2, \sigma_2^2)$ の分布関数を $F(x)$ とおくと，正規分布 $N(\tilde{\mu}, \tilde{\sigma}^2)$ の分布関数 $\tilde{F}(x)$ はある定数 a に対して次の式が成り立つ．$(0 \leq a \leq 1)$

$$x \leq F^{-1}(1-a) \Rightarrow F(x) \leq \tilde{F}(x)$$

ただし，$\tilde{\sigma} = \sigma_1$ かつ，$\tilde{\mu} \geq \mu_1 - \sigma_1\left(\Phi^{-1}(1-a) - F^{-1}(1-a)\right)$ とする．$\Phi(x)$ は標準正規分布 $N(0,1)$ の分布関数である．

定理 4 は正規分布関数の微分をうまく操作することによって証明できるが，証明は紙面の都合から省略する．

定理 4 に沿った方法で近似的に得られた正規分布 $N(\tilde{\mu}, \tilde{\sigma}^2)$ は，⊞ 演算を行うと方針の性質を満たすことができなくなる．⊞ 演算と ⊠ 演算を任意順で繰り返し実行するために次の定理を用いることを考える．

定理 5(⊞ 演算の近似の単調性)

定理 4 において，$\tilde{\mu}$ をできるだけ小さく取った場合の $\tilde{F}(x)$ と $N(\mu_1, \sigma_1^2) \boxplus N(\mu_2, \sigma_2^2)$ の分布関数 $F(x)$ について，$F^{-1}(1-a) - \tilde{F}^{-1}(1-a)$ を σ_1 の関数とみて，$d(\sigma_1) = F^{-1}(1-a) - \tilde{F}^{-1}(1-a)$ とする．すると，$a > 1/2$ の場合は $d(\sigma_1)$ は σ_1 に関して単調増加であり，$a < 1/2$ の場合は $d(\sigma_1)$ は σ_1 に関して単調減少である．

次に述べる手法で ⊞ 演算を近似することにより，近似として得られた正規分布に対して ⊠ 演算を行っても方針の性質を保持できることが定理 5 から言える．パラメータ a が $1/2 < a < 1$ を満たす場合について説明すると，まず，枝重みはそれぞれ正規分布の分散が確定的な枝重みである場合のことを考え，分散が最大となるような全域木 $T_{\max}$ を求めておく．次に，枝重みが確率的な場合に戻って考え，全域木 $T_{\max}$ の重みの分散を $\sigma_{\max}^2$ とおく．演算 ⊠ の計算を近似する際，$N(\mu_1, \sigma_1^2) \boxplus N(\mu_2, \sigma_2^2)$ の代わりに，$N(\mu_1, \sigma_{\max}^2) \boxplus N(\mu_2, \sigma_2^2 + (\sigma_{\max}^2 - \sigma_1^2))$ の近似計算を定理 4 に沿って行う．得られた分布を $N(\bar{\mu}, \bar{\sigma}^2)$ と考え，$N(\mu_1, \sigma_1^2) \boxplus N(\mu_2, \sigma_2^2)$ の近似結果として $N(\bar{\mu}, \sigma_1^2)$ を採用する．パラメータ a が $0 < a < 1/2$ を満たす場合に関しては，$T_{\max}$ の代わりに分散最小の全域木 $T_{\min}$ とその分散 $\sigma_{\min}^2$ を用いる．

3.5 未解決部分

以上により，定理 3 の式から ⊞ 演算の代わりに定理 4,5 による近似計算を行えば，方針の条件を満たす正規分布関数を計算することができるが，定理 3 によれば ⊞ 演算の逆演算を DET の展開時に任意の順番で適用可能でなければならない．しかし，定理 4,5 によって近

似計算を行うと逆演算を方針の条件に合うように定義できるかどうかは本発表時点で未知である．このため，系統的に $\boxminus$ 演算を DET から消去する方法または $\boxminus$ 演算の適切な定義をすることが今後の課題である．

4 結論

枝重みが互いに独立な確率変数として与えられる場合の最小全域木問題に関して，最小全域木の重み分布を行列式の一般化を用いて表現した．

枝重みが正規分布に従う確率変数として与えられる場合，畳み込み積分を避けるために近似手法の設計を試みた．方針として，最小全域木の重み合計が高々 x である確率 $F(x)$ に着目して，与えられたパラメータ a について x が $F^{-1}(1-a)$ よりも大きいところで分布関数の上限を与えることを目標とした．提案手法を完成させるにはあと一段階の工夫が必要であり，近いうちに解決したい． $\uplus$

参考文献

[1] B.Chazzele, “A Minimum Spanning Tree Algorithm with Inverse-Ackermann Type Complexity”, JACM, Vol.47., No.6, November 2000, pp.1028-1047.

[2] V.G.Kulkarni, “Minimal Spanning Trees in Undirected Networks with Exponentially Distributed Arc Weights”, NETWORKS, Vol. 18,1988, pp.111-124.

[3] E. Ando, M. Yamashita, T. Nakata, Y.Matsunaga, “The Statistical Longest Path Problem and Its Application to Delay Analysis of Logical Circuits”, proc.TAU, 2002, pp.134-139.