日本大学経済学部経済科学研究所研究会

【第158回】

2006年11月18日

統計数理システムの解析的研究

日本大学経済学部教授　　大　澤　秀　雄

日本大学理工学部教授　　中　村　正　彰

東海大学理学部教授　　土　井　　　誠

○　大澤

きょうは，日本大学理工学部の中村先生，東海大学理学部の土井先生と私の3人で組みました「統計数理システムの解析的研究」というプロジェクトの成果発表会です．いつもの経済科学研究所の研究会とはちょっと異色なので，聴衆の方もあまりいらっしゃらないとは予想しておりましたけれども，時間になりましたので開始させていただきます．

紹介が遅れましたが，私は経済学部の大澤です．本日の司会進行役も勤めますので，どうぞよろしくお願いします．

まず，簡単に私たちのプロジェクトについて紹介させていただきます．私たちのプロジェクトでは，経済・経営上で起こり得る社会的な現象を確率統計システムあるいは数理システムとしてモデル化し，数理的解析および分析をするということで研究を進めてまいりました．いわば，基礎理論にあたります．この研究所のプロジェクトへの制約では，3人が個々に研究成果をまとめなければならないということでした．そこで，私たちの場合は3人が一つのテーマを共同で研究するのではなく，個々のテーマを研究，その進捗状況を報告，そのうえで問題点解決のために協力しあうという具合に研究を進めてまいりました．

3人がそれぞれに取り組んだテーマは，いずれも経済・経営問題に深く関連したものであります．

土井先生は危険準備金に伴う破産問題，中村先生は取引に伴うオプションに関するファイナンスの問題，そして私は待ちの現象を考察する待ち行列の問題を研究してまいりました．本日の講演は，今お話をした順に行いたいと思います．

では，最初に土井先生のご講演です．講演の題目は，「MEAN RUIN TIME FOR THE RISK RESERVE PROCESS」です．土井先生，よろしくお願いします．

「MEAN RUIN TIME FOR THE RISK RESERVE PROCESS」

土井　　誠

こんにちは．東海大学理学部数学科の土井です．

リスク・リザーブ・プロセス（危険準備過程）と言いますのは，生命保険会社とか損害保険会社などでは保険金の支払いのために危険準備金を積み立てていますけれども，例えば，中越大震災のようなことがありますと，大規模な支払いが生じて会社が潰れることもあります．保険会社に限らず，北海道拓殖銀行は潰れましたし，りそな銀行はいま国営になっています．ですから，潰れないための方策を研究しよう．それには，潰れるまでの平均時間，あるいは破産確率が問題になります．

そこでまず数学的モデルをつくるのですが，危険準備金の積立には上限があります．郵便局の簡易保険などは90兆円の危険準備金を貯めていますが，それ以上貯めると政府の他のところに流用することになりますので，モデルでも上限を設けておこう．破産というのは危険準備金の水準がゼロあるいはマイナスになった状態ですが，大震災など大規模な需要が起きて多額の金額を払わなければいけないような事象はポアソン分布に従って発生します．日本では死傷者が出るような大地震は1.4年に1回ぐらい起きていますが，過去のデータを調べると，それは大体ポアソン分布に従って発生しています．ですから，大規模な需要はポアソン分布に従って発生すると見ていいだろう．

（プリント1）　サンプルパスをお見せしますと，縦軸は時刻tにおける危険準備金の水準で，Lは上限です．横軸は時間で，t_1, t_2, t_3で大震災が起きて大量の保険金支払いが生じたとお考えください．危険準備金は常に積み立てていますので，通常は増加状態にあります．上限に達すると，しばらくその状態を保って，大規模需要が起こるまでは上限に張りついた状態になっている．いまt_3で大規模な需要が発生して水準がゼロになってしまった．ここで破産が起きたということです．政府の管理下に置かれて，ある一定の時間が経てば再び復活する，こういうことを想定しています．

（プリント2）　改めて記号について説明すると，

時刻tにおける危険準備金の水準を$X(t)$とします．$\alpha(x)$は$X(t)=x$での増加率です．保険金を積み立てていますから，危険準備金は常に増加している状態にあるわけです．大規模需要はポアソン過程に従って起きるのですが，そのパラメーターをλとし，分布関数をFとします．いったん上限に張りついたら，大規模需要が起こるまで，そこにとどまっている．

（プリント3）　$X(t, \omega)$は確率過程で，2つの因数を持った変量です．もしωを固定すると，先程の時間tに関するサンプルパスになります．tをもし止めると，典型的な確率変数になります．時間はゼロ以上です．こういう状態を想定しています．

（プリント4）　危険準備金過程というのは，生命保険あるいは介護保険のリスクプロブレム，いわゆる危険過程，それから金融数学，生産・在庫問題，大規模な需要のある生産システム，待ち行列と在庫の絡んだような問題，典型的なM/G/1という待ち行列の問題，こういうところにいろいろ出てきます．これを統一的に捉えて，リスク・リザーブ・プロセス，あるいはストレージプロセスと言いましょうというわけです．

（プリント5）　いままでの歴史ですが，簡単な過程の破産確率については土井，永井，大澤が研究しています．もう少し一般的な入力位相と出力位相があるストレージプロセスについては，その確率については土井と大澤が数値解を求めています．この過程については土井がずっと研究しておりまして，つい先頃，入力位相と出力位相のあるストレージプロセスについての破産確率について結果を得ました．つい先頃得られたばかりですが，それは後ほどお話しします．

（プリント6）　いまわれわれは平均破産時間を問題にしていますので，それを$t(x)$とします．$t(x)$については (1) のような積分微分方程式が得られます．αは水準xでの増加率，Fは大規模需要の大きさの分布関数です．

（プリント7）　これを解きたいのですが，右辺の第1項と第3項に注目してうまく計算してやると，積分したものの微分ですから，先程の (1) の方程式はこのように非常に簡単になります．

（プリント8）　両辺を積分してやればいいんですが，$\alpha(x)$だと積分できないので，$\alpha(x)$は定数と見ていいでしょう．保険金は一定の率で積み立てていることにします．t_Lというのは，危険準備金を一番多く貯め込んでおいて，それから破産するまでの時間ですから，境界条件です．

（プリント9）　xに関して積分すると，こういう積分方程式が得られます．左辺の$t(x)$は右辺の積分の中に入っています．

（プリント10）　第2種のボルテラの積分方程式ですが，これを解くのも厄介なので，確率の世界に持ち込もうというわけで，δという数を導入します．この解は1つしかないと分かっていますので，δについて解けることが分かっています．このδについて解いて，$e^{-\delta x}$をかけてやると，これが密度関数になります．

（プリント11）　そして$t(x)$，$a(x)$などにみんな$e^{\delta x}$をかけていく．それを$t_0(x)$，$a_0(x)$とします．$h_0(t)$を積分しますと$H_0(y)$という分布関数になる．こういううまい処理をしてやると，確率論で出てくる典型的な再生方程式が得られます．これだったら，すでに分かっている手法で比較的容易に解けます．

（プリント12）　どのように解くかというと，これを何回も何回も計算してやればいいわけです．リカーシブルですから，n回繰り返すと，こういう方程式が得られます．H_0^kはH_0のk回の畳み込み，convolutionです．右の項は，後でnを無限大に持っていくと0に収束することが分かります．これを第n項としたときの無限級数はH_0という分布関数のn乗畳み込みの絶対値が0と1の間にある数のn乗より小さいことが分かります．分布関数ですから，もちろん1より小さい．ですからこれは収束する．収束するから，一般項は0に収束することが分かりますので，ここまでのnを飛ばしてこの項がなくなりますから，こうやって解が得られたというわけです．

（プリント13）　これが先程の再生方程式の解です．

（プリント14）　Mは分布関数H_0のn重畳み込みの無限級数です．もちろんH_0^{n*}は$H_0(y)$のn重畳み込みです．

（プリント15）　この一般的な解では分かりにくいので，大規模需要の平均が$1/\mu$の指数分布に従う場合を計算してみます．先程の$M(y)$は無限級数ですが，ここが指数分布であれば比較的簡単

に計算できます．

（プリント16）　$M(y)$ を書き下してみると，またリカンシブルになるのですが，$M(y)$ のラプラス変換をとると，

（プリント17）　こうなりまして，これを解いてやると，これは指数分布だから，M をラプラス変換した結果はこのように非常に簡単になる．これを逆変換してやれば，$M(y)$ は無限級数だったんですが，非常にシンプルなかたちになります．

（プリント18）　$t(x)$，水準が x のときの平均破産時間，水準が x から始まって破産するまでの平均時間はこのように得られます．ここで δ というのは，先程の積分方程式を解けば得られます。α は増加率，λ が大規模地震の発生パラメーター，$1/\mu$ はその大きさの平均，こんな比較的簡単な式が得られます．これは大規模需要の大きさが指数分布だからこそできるわけです．もう少しこれを簡単にして，δ とこの式を代入すると，α と λ と μ の関数になります．

（プリント19）　数値例を挙げますと，上限が10兆円だとします．大規模需要の平均の大きさが10兆円．危険準備金が満杯の状態から始まって破産するまで，t_L を1.0として，これと比較してどのぐらい短くなるかということを言えばいいわけです．先程も言いましたように，日本では死傷者の出るような大規模地震は1.429年に1回起きるということが明治5年以来の気象庁の観測によって分かっています．検定してみると，指数であるということも分かっていますので，このデータを使います．

（プリント20）　増加率を3.0と4.0について計算してみますと，このようになります．危険準備金が満杯の状態に比べて，例えば，8兆円から始めたとすると，増加率が3.0であれば，平均破産時間は0.324．10兆円に比べたら32％しかもたないということです．増加率を3から4に上げると0.5近くなりますので，増加率を1上げただけで相当改善される．増加率を33％上げれば，平均破産時間が53％増えることが分かりました．

（プリント21）　いままでの話は平均破産時間ですが，破産確率はどうか．これは最近得られた結果ですが，P_0 は破産確率，P_L は上限に張りついている確率です．この P_L は確率は全部足すと1ですから，一番下の式で求められます．

（プリント22）　大規模需要の大きさは指数分布に従うとして先程と同じ計算をしますと，危険準備率の初期値 x から始まったときの破産確率はこのように得られました．

（プリント23）　指数分布ですと，先程の平均時間よりはちょっと複雑ですが，なんとか計算できます．

（プリント24）　これも数値例を挙げて計算してみますと，先程と同じように上限が10兆円で，破産してから再び復活するまでの平均時間が0.04の指数分布に従うとして，大規模需要の大きさは5兆円としましょう．

（プリント25）　そうしますと，大規模需要の発生時間間隔の平均が $1/\lambda$ で，増加率を α としたときのグラフはこのようになります．大規模需要の発生は10年間に7回ぐらい起きるので，10/7とします．P_0 が破産確率ですから，破産確率を0.05以下におさめたかったら，α を1.3以上にしないとだめだということが分かります．

（プリント26）　いま，α を1.35だとして，死傷者を伴うような大規模地震が1.47より長ければ破産確率は0.05より低く抑えられますが，大規模需要がもう少し頻繁に起こるようになると，とても0.05にはおさまらないことが分かります．

以上です．どうもありがとうございました．

○　大澤

このお話は次の研究にどういうふうに影響しますか．

○　土井

数学的にはいろいろ問題がありまして，$\alpha(x)$ は水準 x に応じた増加率であるべきなのですけれども，これはなかなか解けない．これをまずコンスタントにしないと解けない．でも，それはなんとか解いてみたい．区分的に水準 x を幾つかの区間に分けて解いて接続すればいいじゃないかと微分方程式の人からは言われていますけれども，そういう問題が1つ．

それから，平均破産時間と確率を出しましたが，これはあくまでも平均時間で，本当は分布が欲しいんです．どのような確率分布で破産するか，破産時間の確率分布が欲しいんですが，それがなかなかできない．

それから，これは時間を十分経過させてしまっているんですけれども，本当は有限時間であるべきですね．北海道拓殖銀行もできて100年も経たないぐらいで，無限に時間が経ったわけではないので，有限時間の確率分布が欲しい．あるいは100年以上破産しないでずーっともちますよ，という確率分布も欲しい．それはなかなか難しくて，いまのところまだできていませんが，そのうちできるだろうと楽観視しています．

それから大規模需要の大きさでは指数分布の場合をやったんですが，いまアーラン分布の場合もやっています．というのは，指数分布は1つでオプションがついていない，生命保険なら生命保険だけですが，普通は生命保険プラス傷害保険とか入院保険をプラスするとかですから独立な確率変数の和ですから，位相が2か3のアーラン分布であるべきなんです．それが難しくて，先程（プリント14）のMの計算がなかなかできない．

いま位相が2のアーラン分布の計算を私のところの大学院生と一緒にやっていますので，もう少しするとできるかもしれません．そうすると，少し現実に近いモデルができるのではないかと思っております．

○　大澤

どうもありがとうございました．

続いて中村先生に「取引費用を伴うオプションの数値解析」についてのご講演をお願いいたします．

「取引費用を伴うオプションの数値解析」

中村　正彰

私からは，先物取引を数学的に扱おうという，いわゆるオプションの話をします．大阪の米相場というのが世界で最初にできた先物取引だそうですけれども，ここで考えている先物取引は，例えば，「10年後にこの値段でこの物を買います」という設定をする．それがVですが，10年後にその額がVよりも高ければ，その人はその差額だけ儲かります．逆に低ければ，その差額だけ損する．そこでこのVをどういうふうに設定したらいいかということです．

数学でも最近，特にここ20～30年，mathematical finance（数理ファイナンス）とかmathematical industrial finance（金融工学）とかやっていますけれども，ぼくの周りでもこういうので実際に投資して儲かったのは1人もいません（笑）．大体大損しているのが多い．ノーベル賞をもらったマーコビッツでさえ，ほとんどかつかつだったようで，「理論と実践は直接関係ない」という笑い話の例の1つにされています．

（プリント2 - 1）　数学の側から見た基本的な設定から話したいと思いますが，このBlack-Schols方程式は資産価格に対する対数正規分布モデルにいろいろな条件をつけて得られた方程式です．VがSに付随するオプションで，Sは原資産の価格，tとTは時間です．定数rとσはそれぞれ無リスク利子率（risk-free interest rate）と資産ボラティリティー（the asset volatility）で，δtはその時間が経ったということです．

もともとのBlack-Schols方程式は，ある時間Tまでの時間のときに，$V_0(S)$というオプションの値段を設定しよう．これから10年後なりにその値段でどうなるか．これはヨーロピアン・オプションといいまして，そのときにその価格で買わなければいけない．アメリカン・オプションは途中でそのオプションを買ったりいろいろやることができるけれども，この場合のオプションは，10年後なら10年後に必ずその額で買いますよ，途中に何もしない，そういうオプションをもとにした方程式です．

数学的にいきますと，ここにSに関する2回微

分がありまして，ここに dv，dt という方程式がある．ちょっとプラス・マイナスがおかしくなっている可能性がありますが，放物型方程式と呼ばれるものです．いまの時間が 0 のところからじわじわ進んでいって，ここに行く．いまがあって将来を予測する，10 年後の値段を決めようという方程式になっていますから，10 年後の値段がこの設定した値段になるように動いてくれればいいわけですけれども，実際にはそうは動かないよというのがこの世の中の話で，そことの差がどうなっているか．これを最初に考えて数学的に扱ったのが Black と Schols という 2 人です．

（プリント 2 - 2）　最初の初期値は 0 にしたかたちで扱うというのが Black-Schols 方程式ですが，これは取引費用 (transaction cost) を入れていないモデルだった．しかし，現実の実務においては途中で取引費用なども入るだろうということで，Hoggard，Whalley，Wilmott によって考えられたのがこの方程式です．先程の Black-Schols 方程式とどこが違うかというと，取引費用から出てくる影響，効果を表すこの項が入っている．

満期データというのは，そのときの値段でそのまま買えるのが一番自然であろう．設定した額と現在の額が差があるというのは，儲け過ぎても面白くない．オプションですから買う方のオプションと売る方のオプションとありますが，そのどちらでもそのときそのときで差が出たのでは面白くない．

非線形方程式を解くときに，そのままでは解けないので，解が凸あるいは下に凸というのを仮定して解析していく．だけど，凹凸を繰り返して動いていくのが普通の解だろう．それではどうしたらいいだろうか．

（プリント 2 - 3）　ここは完全に純粋数学的な微分方程式から来る解析で，経済学的に何を意味するのか私には分かりませんが，S が無限大に行くとき，最後は直線に近いような形になっていく．そういう仮定をつけた初期値を考えると，それに対して先程の方程式は解を持つ．そのときの解は指数オーダーで，ほとんど最後の方は αS と直線に近くなる．

この方程式は，先程言いました $\alpha V=\alpha S$ というのは解になっております．ここは $V=\alpha S$，時間に関して微分なしです．ここも $V=\alpha S$ で，2 回微分なしです．これは 1 回微分して定数になって，ことここが足し合って，ちょうどうまくキャンセルして 0 になって，ここはもちろん 2 回微分なしですから，方程式の解がこういう解になるというのはすぐ分かるわけで，初期値がこれになる．

一方，βS とかこういうのは定常解になっている．だから，解が 1 つではないということも分かるわけで，その辺で話が難しくなる．解がいっぱいあるということは，うまくやると，そこへ行くのに何種類もの道がある．あまりうれしくないのです．最もポテンシャルなども同じで，どんな道を通っても最後は同じですよという，そういうのは逆にうれしいということもあるのかもしれません．

先程の方程式を数値的に扱ってみます．実際に方程式があっても，数学でやる場合は，解があるとかないとかという話と，無限大に行ったときにどうなるかなんていう話で，途中がどうなっているかという話は具体的に分からないことの方が多いわけですが，現実問題としては，ずっと後よりも，ある時間の途中を知りたい．そのためには数値シミュレーションをやるというのがいまの基本的な発想です．

ところが，この HWW 方程式の形を見ると，領域が無限大である．S という値段は 0 以上で無限大ですから，幾らでも高く売れる．時間についても，有限なところにおさまらないで，無限に行ってしまう可能性がある．しかし，実際の計算機は有限の値でしかできないわけですから，無限の値を持つようなものの数値計算はできない．だから，非有界な解は困る．

非線形項絶対値の項があるので，線形でない方程式は面倒くさいというのが普通ですから，そこは何か処理しなければいけない．そのために，解が無限領域であるのを変換して有界領域に直してしまう．非有界な解は有界な解に変換して扱う．非線形項が絶対値の項であるので，スキームは時間に関しては陽スキームで計算する．スキームというのは数値計算をするための計算式ですが，陰スキームというのは，その次の時間も入っている連立方程式を解かないと進まないのですが，陽スキームの方は，ある時間があって，足し算，引き算，掛け算で次のステップへ進められる．工夫としてはこの辺が一番大きいのですが，どういう変

換をやったらいいかというのが1つのポイントになります.

（プリント2 - 4） この方程式の特徴は，あるTという時間が経ったときの値を設定しています．これと同じ形の方程式で有名なのは熱方程式で，針金などの温度分布は時間が経ったらどうなるか．最初の温度はこれですよ，それが時間の経過につれてどう変化するかというのを記述するというのが基本です．ある時間が経ったときの値が与えられて，どういう初期値を持つか，それを求めなさいというのは逆問題で，面倒くさい．特に熱方程式や放物型についてそういうことを考えるのは，問題の設定が適切でなくなっている可能性もある．そういう意味で非常に難しい問題になるけれども，この方程式の場合は問題の本質から，時間を逆にたどっていって初期値を求めようということです．

（プリント2 - 5） 先程の定理で与えられた結果は$V(S, t)$がαSで，時間が経つとSにほとんど等しい．いま$\alpha = 1$で計算しますので，αSにしておきますと，最後はほとんど直線に近くなるだろう．

（プリント2 - 6) そういうことを考えて，いまの変換をしますと，解析的結果で最後はαSになるというので，それとの差を見てみましょう．V_3はそういう方程式です．V_2は時間を逆にたどった方程式で，αSを引くということは，解析から得られたSが無限大に行ったときに答えはαSになるでしょう，αSに近づくのを調べましょう．

ここで注意をしますと，αSとβSですと，$\beta - \alpha$が0ならいいですけれども，0でないとS倍すると無限大になる可能性を消していない．こちらは無限大になる可能性を消しています．こちらがαSになれば，これが$\alpha + S$で終わってしまえば，これは1より小さいわけです．1から1より小さいものを引いているから，絶対値として1より小さい．こちらなら数値計算もうまくいくだろう．ただ，後で紹介させていただきますが，方程式そのものはこういう変換をすると見た目がものすごく汚いし，面倒くさい．

これはαSに近づいてくれるという仮定のうえで有界になりますけれども，そうでなかったら分からないわけです．方程式はそういうのはあると言っているけれども，1じゃないので，他の解がある可能性もあるので，うまく何かやりたい．

（プリント2 - 7） そこでもう1回方程式を変換していきますと，V_3に対する変換方程式はこういう方程式，V_4に対する変換方程式はこういう方程式が得られます．見ていただくと分かるように，この辺にいやな項がいっぱい出てきています．この辺は大したことないんですけれども，ここに面倒くさいなあというのが出てきているし，絶対値がついているところ，非線形の中身が増えている．ですが，まあこんなものならこういう解で計算できる．

（プリント2 - 8） 無限領域のままではまずいと先程言いましたが，それに対してはこういう変換をしますと，時間とSに対する方程式ですから，Sをこういう形で変換してxに直してしまいます．そうすると，Sが$-\infty$から∞のときは区間$(-1, 1)$上に，0は0に変換されます．

（プリント2 - 9） 具体的には，例えば，このようにして逆に計算すると，これになる．微積分の変数変換はこんなかたちになって，実際に得られた方程式がV_3に対する方程式で，それにこういう変換をやったのはこんな方程式になります．tとτはちょっと変えましたが，こんなかたちで，この中に非線形をこうやって，ここに2回微分が入っているし，ここが2乗ですし，この辺もあって，こんなかたちのあまりうれしくない，やはり非線形項です．スターはこれとこれを掛けるということです．

（プリント2 - 10） V_4については，これも面倒くさそうな方程式で，やっぱり非線形項がここについているし，2回微分がここにもついているし，ここにもついている．ただ，この辺とこの辺が変数係数ですが，大したことはなくて，計算する上ではここが最もネックになって一番面倒くさい，そういう方程式です．

（プリント2 - 11） 実際にこの数値シミュレーションをやったときのパラメーターです．αは1ですから，単純にBlack-Scholsの最初のSのオプションの価格とそのときの値段というのは大体同じになってくれるのが一番自然であろう，そういう仮定です．この辺は，これが必ずいい値かどうか分かりませんが，こういうときに解がありますよとやった定理を満たすようにつくっている．$\varepsilon = 1$．これは何でもよくて，適当につくっている．そして初期値$V_0(S) = S \cdot \tanh(S/2)$ですので，最

後はSに無限に近くなるような初期値になっています．こちらもそうで，どっちの初期値でもSに行くように設定してあります．

（プリント2 - 12）　計算法ですが，時間に関してはEulerである．もとのtというステップからΔt時間だけ経った後の値計算はこういうかたちです．FDMというのは有限要素法（Finite Difference Method）という意味で，2回の微分係数についてやりますと，こういうかたちである．これは誤差のオーダーでいきますと，打ち切り誤差というのは2乗になる．方程式の言語はFortranである．精度は倍精度ですから，有効数字7～8桁あると思っていいわけです．そういう計算でやる．

（プリント2 - 13）　時間は$T=50$に設定しておきます．Sが大きくなりますと，最後はほとんどSと等しくなっているようなかたちです．先程の初期条件との関係ですと，Sはxについて-1から1に変換しています．それを1/200にします．だから-1から1を200等分している．そうしますと，先程のスキームとの関係で，Δtはこれの2乗の2分の1以下でなければいけない．そうでないと安定性が壊れるという基礎理論がありますので，これは10^{-4}分のなんとかになりますが，そうして計算したものが，こちらがV_2で，こちらがu_3です．これがいいかどうか，得たい値かどうか分からない．

（プリント2 - 14）　これは誤差を並べたもので，$\tau=0$．最初のときの誤差はもちろんほとんどなしで，tをどんどん進めていったときの誤差を見るわけですが，こちらが0.004とか，こちらが001で，τを大きくするとだめだということです．どんどん時間を大きくとっていくと，時間が経ったときにあまりいい値になってくれていない．τが時間が大きくなっていったら最後は$=0$になってほしいわけですが，これを見ていただくと分かるように，最初のうちはτが0.1ということは誤差はものすごく小さいですね．だんだんこれが上に行くということは，誤差が大きいということです．どんどん大きくなって，時間が1.0経ったときは誤差がすごく大きい．τを50にしたら，もっと大きくなる可能性がある．

この計算は，実を言うとあまりうまくいっていない．先程言いましたが，有効数字7～8桁ありますので，それで計算すると誤差は大体10^{-5}ぐらいにはならないと実際の計算としてはだめなわけです．なぜこれがだめなのかは分かりません．

先程のV_4についての方程式もこんなかたちでやりますと，初期値はほとんど4より先は一致している．ところが，誤差を見ますと，τが大きくなったときに誤差がどんどん大きくなっている．どうもうれしくないというのが1つです．

どうしてこうなるのかという，最後の結論と今後の問題です．

まず，ホガード・ワーレイ・ウィルモット方程式の理論的な結果，つまり定理の主張が実際に起こるのか，シミュレーションで実際にどんな解になっているのか調べようというのが最初の目的で，無限領域における計算を，手法としては倍精度，陽的オイラー法で解いたんですが，誤差が最後にかなり大きく残っている．どうもそれはうれしくない結果で，定理の主張を確実に裏付ける結果にはなっていない．つまり，Sに近づいていない可能性があるわけです．Sに近づいていれば最後誤差がほとんど0にならなければいけないのが，そうはなっていない．

ただこれは，先程も言いましたが，放物型の方程式で逆問題を解くというのは，ある意味ではタブーに近い部分も持っている．つまり非適切な可能性がある．問題に解が必ずあるという保証がない問題を解いている．それが1つと，数値的に解く場合に，非線形性がありますので，われわれが使ったスキームが本当にいいスキームかどうか分からない．非線形ではこうやればいいという定番はないので，試行錯誤になっていく．そこで誤差がたまっていて，答えが得られていない．これは数値計算の部分の問題です．

もう1つは，$\kappa=0$の場合，つまり右辺の非線形がないBlack-Schols方程式も具体的に計算をして，トランザクションコストが入ったときに途中でどういう違いがあるのか比較する必要があるだろう．将来の問題として，これはそのうちやる予定です．

なお，今回の発表では私1人の名前にしておきましたが，実際は一橋大学の石村直之さん，徳島大学の今井仁司さんの協力も得ています．

どうもありがとうございました．

「Random Arrival Acceptance Windows をもつ待ち行列について」

大澤　秀雄

私から「Random Arrival Acceptance Windows をもつ待ち行列について」発表させていただきます．「待ち行列」という言葉自体，あまりお馴染みがないかもしれませんので，待ち行列とはどういうものかということからお話しさせていただきたいと思います．

前のお二方が数式的に厳密なことをやられましたので，ここではあまり数式を使わないでお話しさせていただきますが，待ち行列というのは，銀行とか郵便局，駅のみどりの窓口とか，何かサービスを受けるために行列をつくるという現象で，現実的にいろいろなところで起こり得るものです．スーパーのレジなどもそうですし，医療機関における待合室もそうで，特に日本の医療機関では，長時間待たされて，診察時間はわずかというようなことも多々あるわけです．また，事務処理を行なう書類なども，机の上に並べておけば，それは待ち行列になります．コンピューターのタスク処理を行なう際に，コンピューターの中でも待ち行列は起こっている．いろいろなところで待ち行列の現象は起こっていますが，その混雑を起こす原因を探って，それをどうするかということでシステムの精度評価にもつながってくる問題でもありまして，この待ち行列を英語ではQueueとかQueueingあるいはQueuingと言います．

なぜ待ち行列ができるか，簡単に模式化してお話ししますと，サービスを受けようとする人が到着して，窓口あるいはサーバーがありまして，この四角の中にある○の人がサービスを受けて，そして退出する．この人が出ると，次の人が入ってサービスを受けるわけですけれども，窓口でサービスを受けている状態が長く続くと，行列ができてくる．このように待ち行列というのは基本的に3つの流れがあって，到着，それからサービスとか窓口と呼ばれるもの，そして退去・退却，こういう現象で起こるわけです．いま単純なモデルでお話ししましたけれども，ネットワークなど，複雑な待ち行列も含めて，いま盛んに研究が行なわれています．

なぜ行列ができるのか考えてみますと，到着というのは必ずしも規則的に来るわけではない．もちろん規則的に来るような現象もありますけれども，到着の時間の間隔をとらえたときに，一般には不規則にランダムにやってくる．待ち行列の理論ではこれを確率変数としてとらえて，確率過程として定式化することが行なわれます．

先ほど述べたように，到着においてランダムな現象が起こってくるわけですけれども，サービスの面，実際に処理をする面においても，サービス時間が不規則であるということがあります．これもサービスの時間間隔をランダムな現象としてとらえて，数学的に表します．

到着とサービスにおいて，2つのランダムな現象が絡み合ってくる．これが輻輳現象，混雑の現象を起こす原因ともなるというわけで，いろいろ研究がされてきています．待ち行列の理論ができてから1世紀近く経っていますので，単純なモデルについては研究し尽くされていますけれども，きょうは，通常の待ち行列に対して，到着の流れを制御することを考えてみたい．そういうモデルを提示して，その解析結果とシミュレーションを通した分析についてお話ししたいということです．

通常待ち行列というのは，A/B/cという記号で表します．ここで，Aは到着の時間間隔を表す記号で，Bはサービスの時間の間隔を表す記号，cはサーバーの人数とか窓口が幾つあるかという記号として使っています．

（プリント3 - 1）　模式図で行列を起こす現象を見てみますと，t_{n-1}のところに3人いて，線上の緑の人がサービスを受けているという状況を表しています．この時点で黄色の人が到着した．緑の人はサービスを終えて出ると，順番に前へ詰めていく．次のt_nの時点で赤い人が到着した．ここで青い人が終わったけれども，次の人がなかなか来ないと，行列は減っていくことになります．このまま来ないと，赤い人も終わって，行列は空になる．アイドル状態と言いますけれども，こういうことも起こり得ますが，ここに新たな到着が起これば，またどんどん行列はできてくる．こんな具合で行列は変動していきます．

（プリント3 - 2）　Tは到着の間隔を表しますが，到着の間隔がこのように長くなったり短くなったり，ランダム到着という現象が起こります．

そのことを到着率λで表しますが，平均の到着間隔はこの逆数で$1/\lambda$になります．この到着率は，単位時間当たりにλ個の到着があるというような感覚でとらえていただけばいいと思います．そしてこのランダムな現象はサービスの方にも起こります．Sはサービスの時間を意味しますが，これがサービスを受ける人の要請で，短くなったり長くなったりする．これをサービス率μで表します．単位時間当たり平均的にμ個のサービスができるということを表すパラメーターです．平均サービス時間はその逆数の$1/\mu$になります．

到着とサービスのランダム性によって行列が起こり得る可能性があるわけですが，到着が頻繁に起こると，システムがもたないくらいに行列ができてしまうという現象も起こり得る．そこで，到着を少し制御して，混雑現象，輻輳現象を抑えられないかということで考えたのが，きょうの題名にもあります，Random Arrival Acceptance Window (RAAW) Queuesモデルです．

到着のところにウインドウを設けて，それを開いたり閉じたりして客の流れを制御する．ウインドウが開いているときはお客さんを受け入れることができて，閉じていると，その場に来たお客さんは中に入ることができないというモデルを考えるわけです．しかも，ウインドウが開いている間に受け入れることのできる数の上限を設けて，その受容数に達するとウインドウをクローズして一定期間受け入れることができないようにする．

この待ち行列系を表す記号として，$A|a_m/B/c$を使います．ここで，Bとcは先程と同じ意味ですけれども，Aはウインドウオープンの時間間隔．つまり，ウインドウが開いて，次の時点で開くまでの時間の間隔を表す記号とします．aはウインドウが開いてからお客さんがどういう流れで来るかということを表す記号で，遅れ到着時間と呼ぶことにします．mは1つのウインドウが開いている間に何人まで受け入れられるかという受容数です．ウインドウが開く時間の間隔の平均，つまり平均ウインドウサイズを$1/\lambda$とします．平均の遅れ時間を$1/\nu$，平均サービス時間を$1/\mu$とします．

（プリント3 - 3）　こういう設定のもとで，RAAWシステムを模式的に表しますと，t_nという時点でn番目のウインドウが開いたとします．そして次のt_{n+1}の時点で$n+1$番目のウインドウが開く．このt_nとt_n+1の間がウインドウが開く時間間隔になります．この間に受容数m人だけ受け入れることができる．お客さんがどんどん来て，ここまでは受け入れることができるわけですけれども，m人のお客さんが来たところでウインドウをクローズします．そうすると，その後に来たお客さんはリジェクトされてしまう．一般的なmでやれればいいのですが，ここではmは1，つまり1つのウインドウの間で1人しか受け入れられないという簡単なモデルを考えたいと思います．

まず，従来の待ち行列のモデルとRAAWモデルとの比較をシミュレーションで確認してみたいと思います．通常のM/M/1と書かれていますけれども，このMという記号は指数，ポアソンということですけれども，非常にランダム性のある分布と考えられています．このM / M /1待ち行列において，パラメーターλを0.97，μを1.0としますと，平均到着時間は逆数となり1/0.97ですから，計算すると1.03です．平均サービス時間は，μが1ですから，そのまま1ですね．これは待ち行列ではヘビートラフィックの方に入るもので，混雑が非常に起こりやすい現象と考えられます．

（プリント3 - 4）　この場合のシミュレーション結果をグラフに表しますと，横軸が時間，縦軸が系内に行列をつくっている人数です．実際にはこんなことは起こらないと思いますけれども，お客さんが途中で逃げないと仮定すると，最後の方はもうシステムがもたないぐらいにお客さんが並んでしまうという現象が観察されます．

それに対して，RAAWというシステムを考えて，流れを少し制御するとどうなるか．λとμは同じですが，ウインドウの開く間隔をお客さんの流れに合わせて同じパラメーターで置きます．実際のお客さんはそこから少し遅れてやってくるというのが$\nu = 3$という数値で表されています．平均の遅れが0.33ですから，窓が開いてちょっと経過してから来る，こういうパラメーターでシミュレーションします．

（プリント3 - 5）　その結果を見ますと，非常に行列が解消されていることが一目で分かります．先程のは上限80で，実際に非常に混雑現象が起きていましたけれども，これは上限30になっていて，最大でも10人前後の客の行列しかできていない．先程の状況と全く同じお客さんの流れを

想定して，RAAWのシステムを考えたシミュレーションの結果として，非常に行列が緩和されているわけです．

この実験を通しても，RAAWというシステムは非常に効果があるのではないかと思われます．例えば，歯医者の指定到着などはいい例でありまして，歯医者は何時までに来なさいと患者に到着を指定します．その結果，患者はあまり待たなくて済むことになりますので，このモデルは非常に効果を持つことが分かると思います．

次に，窓を開ける時間をどのように設定したらいいか，Optimal Window Sizeということを考えてみます．ウインドウのサイズを決めるのに，どういう条件のもとで最適なのか，が問題になります．そのために，ここではexpected rewardを導入します．これはシステム側から見た報酬を表す式で，$r(\Lambda)$は1つのウインドウで平均的にどのぐらいシステムが報酬を持てるか，利益があるかということを表します．C_sは平均の利益で，サービスによってシステムの受けることのできる単位時間当たりの利益です．C_wは客の待ち時間に応じてシステムが被る単位時間当たりのコストです．待ちが長くなると，それだけシステム側が料金を支払わなければいけないということが考えられますので，単位時間当たりの待ちに対してコストがかかる．C_lはロスコストです．全部の客を受け入れられるわけではないので，受け入れることができなかったお客さんに対して損失を受ける．システム側としてはそこで利益が出なかったわけですから，それを機会損失ととらえて，C_lと置きます．Λは$1/\lambda$で，平均ウインドウサイズを表しますが，このΛを幾つにしたらいいのかということがいま問題になるわけです．p_aは受容確率で，実際に受け入れることのできる確率です．W^*は平均待ち時間で，これはシンプルなモデルに対しては解析的に求めることができます．

（プリント3 - 6）　ここでは，リウォードファンクション(Reward Function)を導入しまして，これを最大にするという意味での最適モデルを考えます．1つのウインドウからウインドウの間に得られるリウォード$r(\Lambda)$に対して，T_cはシステムの稼働の全時間，λT_cはシステムが稼働している間に何回ウインドウを開くことができるかということを表す平均です．それと1つのウインドウの間に得られるリウォード$r(\Lambda)$を掛けることによって，システムの全稼働時間の間にどれだけのリウォードがあるかということを表す関数$R(\Lambda)$になります．

一般的に調べて最適なΛの存在を示せばいいのですが，まだそこまで至っておりませんので，数値例によって最適なΛが実際に存在するのかどうかを見ました．実際のモデルがあるということではなくて，数値例としてたまたま頭で考えたにすぎませんが，基本的に平均のサービス時間を単位時間として考えて，こういうパラメーターを設定します．

（プリント3 - 7）　このパラメーターのもとで数値計算を行ないまして，平均ウインドウサイズに対するリウォードファンクションの動きを計算した結果がこのグラフです．横軸が平均のウインドウサイズで，縦軸はリウォードです．D|Mと書かれていますけれども，これはウインドウを一定時間に置いた場合の数値例です．その場合にリウォードが最大になるような平均時間というのは，存在することがこの実験で確認されております．それもただ1つに決まっている．

（プリント3 - 8）　窓を開ける時間をランダムに設定しても同じような現象が起こりまして，これはまた別のモデルですけれども，こういうグラフを得ることができます．

だから，いろいろなモデルに対しても似たような結果か得られ，確かにオプティマルなウインドウサイズはただ1つに決定しそうだ．これはまだ解析的に確認されているわけではないですけれども，数値例としてはオプティマルなウインドウサイズが存在しそうだ．ただ1つにありそうだということが予測されます．

（プリント3 - 9）　オプティマル・ウインドウ・サイズの数値例ですが，いずれのモデルも，上の方の例だと，来たお客さんをそれほど拒否しなくて済んで，しかも最大利益を得られるというような結果が得られています．

時間的にだいぶ経ちましたので，以上で発表は終わらせていただきます．ご静聴ありがとうございました．