ON DAVIDSON'S PROBLEM IN THE COLLECTIVE RISK THEORY

Elena Cardona*

Abstract. In this paper Davidson's classic problem concerning the solution of an integro-differential equation regarding the collective risk theory with the aim of examining the probability of the failure of an insurance company is further analysed. The validity of a new representation's formula to the solution of the problem is demonstrated after having discussed the question of the existence of that solution.

Keyword. Ruin probability. Integro-differential equation.

1. Introduction.

As it is well known, the collective risk theory, introduced by Lundberg and subsequently developed by various authors during the last hundred years, has been a fundamental contribution to questions concerning the probability of the failure of an insurance company in a finite time.

The usual approach to such problems consists in examining the dynamic over time of the risk reserve's fund, which the company assigns in the starting time to the management of non-life insurance portfolio with homogeneous policies covering repeatable accidents. The topic will now be briefly reviewed in order that the problem in question can then be discussed.

2. Recalls on the collective point of view

With reference to the period $(0,t)$, $t \in R^+$, we put

$W(t)$ = size at time t of the risk reserve's fund, which an insurance company above specified assigns to whole portfolio or its part; in particular: $W(0) = x$.

Z_i = random variable (r.v.) " Company's outlay relative to i-th claim";

* Department of Mathematics and Statistics, University "Federico II", Naples (Italy). E-mail: cardona@unina.it

Work carried out within the framework of the research project CNR00C91C_006 funded by CNR Agenzia 2000.

c = flow of the company's receipts without fund's yield, assumed constant over time (Lundberg's hypothesis, 1903);

N(t) = r.v. "number of claims in (0,t)";

S(t) = r.v. "Company's total outlays relative to *N(t)* claims".

Therefore

$$S(t) = \sum_{i=1}^{N(t)} Z_i \tag{2.1}$$

$$W(t) = x + ct - S(t) \tag{2.2}$$

In the sequel we assume the r.v. Z_i independent and identically distributed (i.i.d.) with c.d.f. *P(z)* absolutely continuous, and therefore with p.d.f. *p(z)* continuous, in R_0^+.

With these hypotheses and positions, the two factors which determine the net risk premiums flow are:

i) the average outlay per claim, given by

$$\mu_1 = E(Z_i) = \int_0^{+\infty} z\, dP(z) = \int_0^{+\infty} z\, p(z) dz, \quad \forall i \tag{2.3}$$

ii) the average number $E[N(1)]$ of claims in the unit of time, which, in the hypothesis that *N(t)* follows Poisson distribution with parameter $\nu t = E[N(t)]$, is given by intensity ν.

In the same hypothesis, the r.v. *S(t)* follows compound-Poisson distribution with intensity ν for the arrivals process and its expected value is given by

$$E[S(t)] = \mu_1 \nu\ t \tag{2.4}$$

Owing to loading on the premiums with mean rate $\lambda > 0$, the parameters must be fixed in such a way that

$$c = \mu_1 \nu\ (1+\lambda) \tag{2.5}$$

For the sake of exposition's simplicity, the unit of time (or operational time if the arrivals process were non-homogeneous) will be chosen in such a way that $\nu=1$. Moreover, one will be assumed that the average outlay per claim is the unit of amounts, so $\mu_1 = 1$. Therefore (2.5), giving $\mu_1 \nu = 1$, becomes

$$c = 1 + \boldsymbol{l} \tag{2.5'}$$

In the aforementioned paper [4] Davidson dealt with fundamental questions concerning the theory of risk and of ruin in the hypothesis that the safety loading is variable in function of the initial level x of the risk reserve's fund.

He introduces the following values:

$\psi(u, x)$ = probability that the fund initially at x level falls below $x-u$ $(u \geq 0)$;

$\chi(u, v, x)dv$ = probability that the fund, initially at x level, falls below $x-u$ $(u \geq 0)$ and that, when it does so for the first time, its value is between $x-(u+v)$ e $x-(u+v+dv)$.

Therefore, $\forall\, u \geq 0$ it results

$$\psi(u, x) = \int_0^{+\infty} \chi(u, v, x)\, dv .$$

That given, Davidson analyses all the mutually exclusive events whose probability is $\chi(u, v, x)dv$. By means of differential arguments, he obtains the equation

$$\frac{\partial}{\partial u}\chi(u, v, x) - \frac{\partial}{\partial v}\chi(u, v, x) = \chi(u, 0, x)\ \chi(0, v, x-u) \tag{2.6}$$

By evaluating the risk reserve through the various possibilities about the number of claims in $(0,t)$, Davidson demonstrates that the process is regulated by the following differential equation

$$\frac{\partial}{\partial x}\nu(v, x) + \frac{\partial}{\partial v}\nu(v, x) + \left[\nu(0, x) - \frac{1}{1+\lambda(x)}\right]\nu(v, x) + \frac{1}{1+\lambda(x)}\, p(v) = 0 \tag{2.7}$$

where $\eta(v, x) = \chi(0, v, x)dv$ and being $\lambda(x)$ the safety loading rate, which is supposed a function of the initial level x of the risk reserve.

Laurin and Lundberg (see [11], [14]) had already obtained (2.7) via other methods. From (2.7) the integro-differential equation in the unknown $\psi(u, x)$

$$[1+\lambda(x)]\,[\psi_u(u, x) + \psi_x(u, x)] - \psi(u, x) + \int_0^u \psi(u-z, x-z)\, p(z)\, dz + \\ + 1 - P(u) = 0 \tag{2.8}$$

follows.

The (2.8), with the positions

$$\xi = x - u \tag{2.9}$$

$$\overline{\psi}(u,\xi) = \psi(u,\xi+u) \tag{2.10}$$

becomes

$$\overline{\psi_u}(u,\xi) = \frac{1}{1+\lambda(\xi+u)}\left[\overline{\psi}(u,\xi) - \int_0^u \overline{\psi}(z,\xi)\cdot\cdot p(u-z)\,dz - 1 + P(u)\right] \tag{2.11}$$

that Davidson, in the aforementioned work, resolved by means of a procedure based on the theory of integral equations, assuming the initial condition

$$\overline{\psi}(0,\xi) = \psi(0,\xi) \tag{2.12}$$

The (2.11) with the condition (2.12) is often cited in the literature as "Davidson's problem".

Let us remark that, put

$$\overline{\psi}(u,\xi) = 1 - f(u,\xi) \tag{2.13}$$

(2.11) leads to

$$f(u,\xi) - \left[1+\lambda(u+\xi)\right] f_u(u,\xi) = = \int_0^u f(u-z,\xi)\,p(z)\,dz \tag{2.14}$$

with the initial condition

$$f(0,\xi) = 1 - \overline{\psi}(0,\xi) \equiv f_0 \tag{2.15}$$

in which f_0 is a constant suitably assigned .

Moreover, the assignment of f_0 gives rise to some difficulties. Really, remembering (2.9) and the meaning of $\psi(u,x)$, if one puts $u = x$ or $\xi = 0$, $\psi(u,x)$ signifies the asymptotic probability of ruin in proper sense, or rather that the risk reserve, initially at x level, will sooner or later fall to zero. In such a case, let us write ψ in the form $\psi(x,x) = \psi^*(x)$ and put $f(x) = 1 - \psi^*(x)$, that is the asymptotic probability of non-ruin when x is the initial fund. That stated, it results that the constant f_0, which appears in (2.11), supposing a variable loading, must be fixed in such a way as to satisfy the condition: $f(+\infty) = 1$ (obviously non-ruin is assured if the initial reserve is infinitely large). Due to this problem, a resolving procedure has not yet been found in the case of an infinitely large initial fund. Algorithms of asymptotic calculus of the constant f_0 can, however, be applied with reference to the similar problem $f(k) = 1$, for a sufficiently large k (see [2], [10]).

3. New thoughts on Davidson's problem.

Let us consider the following integro-differential problem

$$\begin{cases} f(u,\boldsymbol{x}) \; - \; \left[\, 1 + \boldsymbol{l}(u+\boldsymbol{x}) \,\right] \dfrac{\P\, f(u,\boldsymbol{x})}{\P\, u} \; = \displaystyle\int_0^u f(u-z,\boldsymbol{x})\, p(z)\, dz \\ \qquad f(0,\boldsymbol{x}) \; = f_o \qquad\qquad\qquad\qquad f \in C^1(R_0^+) \end{cases} \tag{3.1}$$

where: $0 < f_0 < 1$, $C^1(R_0^+)$ is the class of continuous functions with continuous derivatives in R_0^+ e $P(z)$, $p(z)$ are defined as in § 2. Besides, it results:

$$\int_0^\infty \left[\, 1 - P(z) \,\right] dz \; = \; 1$$

taking as amount's unit the mean outlay per claim.

It is known that the integro-differential problem (3.1) admits only one solution. With the parameter ξ fixed, the existence of the solution can be demonstrated by using the successive approximation method, which however allows to find a representation's formula for the solution, performed in the following § 4. To this aim, we observe that the first of (3.1) yields:

$$\frac{\P f(u,\boldsymbol{x})}{\P\, u} = \frac{1}{1 + \boldsymbol{l}(u+\boldsymbol{x})}\, f(u,\boldsymbol{x}) \; - \; \frac{1}{1 + \boldsymbol{l}(u+\boldsymbol{x})} \int_0^u f(u-z)\, p(z)\, dz \tag{3.2}$$

which, integrated between 0 and u, becomes:

$$\begin{aligned} f(u,\xi) = f(0) + \int_0^u \frac{f(\tau,\xi)}{\left[1 + \lambda(\tau+\xi)\right]}\, d\tau + \\ - \int_0^u \frac{1}{\left[1 + \lambda(\tau+\xi)\right]} \int_0^\tau f(\tau - z, \xi)\; p(z)\; dz\, d\tau. \end{aligned} \tag{3.3}$$

Putting:

$$f_0 = f(0,\xi) \tag{3.4}$$

$$f_n\,(u,\xi) = f(0,\xi) + \int_0^u \frac{f_{n-1}(\tau,\xi)}{\left[1 + \lambda(\tau+\xi)\right]}\, d\tau \; +$$

$$-\int_0^u \frac{1}{[1+\lambda(\tau+\xi)]}\int_0^\tau f_{n-1}(\tau-z,\xi)\ p(z)\,dz\,d\tau\,; \qquad (n=1,2,3,...) \tag{3.5}$$

$$\beta_n(u,\xi) = f_n(u,\xi) - f_{n-1}(u,\xi)\ ; \quad (n=1,2,3,\dots) \tag{3.6}$$

$$\beta_0 = f_0\ . \tag{3.7}$$

one obtains

$$f_n(u,\xi) = f(0,\xi) + \beta_1(u,\xi) + \beta_2(u,\xi) + ... + \beta_n(u,\xi)\ ;\ (n=1,2,3,\dots) \tag{3.8}$$

and also

$$\beta_{n+1}(u,\xi) = \int_0^u \frac{\beta_n(\tau,\xi)}{[1+\lambda(\tau+\xi)]}\,d\tau - \int_0^u \frac{1}{[1+\lambda(\tau+\xi)]}\cdot\int_0^\tau \beta_n(\tau-z,\xi)\ p(z)\,dz\,d\tau\ ;$$
$$(n=1,2,3,\dots) \tag{3.9}$$

We now prove that:

$$\beta_n(u,\xi) = f_0\int_0^u \frac{1}{1+\lambda(\tau_1+\xi)}\cdot\int_0^{\tau_1}\frac{1}{1+\lambda(\tau_2+\xi)}\cdot\dots\cdot\int_0^{\tau_{n-2}}\frac{1}{1+\lambda(\tau_{n-1}+\xi)}\cdot$$
$$\dots\cdot[1-P(\tau_1-\tau_2)]\,d\tau_n\cdot d\tau_{n-1}\cdots d\tau_1 \tag{3.10}$$

Dim.:
Proceeding by induction, for n=1 one obtains

$$\beta_1(u,\xi) = \int_0^u \frac{f_0}{1+\lambda(\tau_1+\xi)} - \int_0^u \frac{1}{1+\lambda(\tau_1+\xi)}\int_0^{\tau_1} f_0\ p(z)\,dz\,d\tau_1 =$$

$$= f_0\int_0^u \frac{1-P(\tau_1)}{1+\lambda(\tau_1+\xi)}\,d\tau_1 \tag{3.11}$$

being $P(0) = 0$.

Let us now verify that (3.10), supposed to be true for $n=k$, is also valid for $n=k+1$. In fact :

$$\beta_{k+1}(u,\xi)=\int_0^u \frac{\beta_k(\tau_1,\xi)}{1+\lambda(\tau_1+\xi)}d\tau_1-\int_0^u \frac{1}{1+\lambda(\tau_1+\xi)}\int_0^{\tau_1}\beta_k(\tau_1-z,\xi)\,p(z)\,dz\,d\tau_1=$$

$$=f_0\int_o^u \frac{1}{1+\lambda(\tau_1+\xi)}\int_0^{\tau_1}\frac{1}{1+\lambda(\tau_2+\xi)}\int_0^{\tau_2}\frac{1}{1+\lambda(\tau_3+\xi)}\cdots\int_0^{\tau_{k-1}}\frac{1}{1+\lambda(\tau_k+\xi)}\cdot$$

$$\cdots\left[1-P(\tau_2-\tau_3)\right]d\tau_{k+1}\,d\tau_k\cdots d\tau_1\;\;d\boldsymbol{t}_{k+1}\,d\boldsymbol{t}_k\cdots\;d\boldsymbol{t}_1\;\;-$$

$$\int_0^u\frac{d\boldsymbol{t}_1}{1+\boldsymbol{l}(\boldsymbol{t}_1+\boldsymbol{x})}\int_0^{t_1}p(z)\int_0^{t_1-z}\frac{\mathrm{d}\boldsymbol{t}_2}{1+\boldsymbol{l}(\boldsymbol{t}_2+\boldsymbol{x})}\int_0^{t_2}\ldots\ldots \qquad (3.12)$$

$$\int_0^{\tau_k}[1-P(\tau_{k+1})]\prod_{i=3}^{k+1}\frac{1}{1+\lambda(\tau_i+\xi)}\prod_{j=2}^{k}[1-P(\tau_j-\tau_{j+1})]\;d\tau_3\ldots d\tau_{k+1}\Big\}$$

As a consequence of

$$\int_o^{\tau_1}p(z)\int_o^{\tau_1-z}\left\{\frac{1}{1+\lambda(\tau_2+\xi)}\int_o^{\tau_2}\frac{1}{1+\lambda(\tau_3+\xi)}\cdots\int_o^{\tau_{k-1}}\left[1-P(\tau_k-\tau_{k+1})\right]\cdot\right.$$

$$\left.\cdot\left[1-P(\tau_{k-1}-\tau_k)\right]\cdot\cdots\left[1-P(\tau_2-\tau_3)\right]d\,\tau_{k+1}\cdot d\tau_k\cdots d\tau_3\;\right\}d\tau_2\,dz=$$

$$=\int_0^{\tau_1}\left\{\frac{1}{1+\lambda(\tau_2+\xi)}\int_0^{\tau_2}\frac{1}{1+\lambda(\tau_3+\xi)}\cdots\int_o^{\tau_{k-1}}\left[1-P(\tau_k-\tau_{k+1})\right]\cdot\right.$$

$$\left.\cdot\left[1-P(\tau_{k-1}-\tau_k)\right]\cdot\cdots\left[1-P(\tau_2-\tau_3)\right]d\tau_{k+1}\cdots d\tau_3\;\right\}\int_0^{\tau_1-\tau_2}p(z)\,dz\,d\tau_2 \quad (3.13)$$

we obtain

$$b_{k+1}(u,x)=\;f_0\left\{\int_0^u\int_0^{t_k}\ldots\int_0^{t_k}[1-P(t_{k+1})]\;\prod_{i=1}^{k+1}\frac{1}{1+l(t_i+x)}\;\cdot\right.$$

$$\left.\cdot\prod_{j=2}^{k}[1-P(\tau_j-\tau_{j+1})\;\;]\,d\tau_{k+1}\,d\tau_k\ldots d\tau_1+\right.$$

$$\left.-\int_0^u\int_0^{\tau_1}\ldots\int_0^{\tau_k}[1-P(\tau_{k+1})]\;P(\tau_1-\tau_2)\prod_{i=1}^{k+1}\frac{1}{1+\lambda(\tau_i+\xi)}\cdot\right.$$

$$\left.\cdot\prod_{j=2}^{k}[1-P(\tau_j-\tau_{j+1})]\;\;d\tau_{k+1}\,d\tau_k\ldots d\tau_1\right\}=$$

$$=f_o\;\int_o^u\frac{1}{1+\lambda(\tau_1+\xi)}\;\int_0^{\tau_1}\frac{1}{1+\lambda(\tau_2+\xi)}\;\int_o^{\tau_2}\frac{1}{1+\lambda(\tau_3+\xi)}\cdots$$

$$\prod_{j=1}^{k}\left[1-P(t_j-t_{j+1})\right]d\tau_{k+1}\,d\tau_k\cdots d\tau_1\;. \qquad (3.14)$$

Let us now prove

$$\beta_n(u,\xi) \le f_0 \frac{u^n}{n!} \qquad (n=1,2,3,\ldots)$$

(3.15)

Dim:

Proceeding also here by induction's process, owing to (3.11) it results

$$\beta_1(u,x) \le f_0 \int_0^u \frac{1}{1+\lambda(t_1+x)}\,dt_1 \le f_0\,u \qquad (3.16)$$

Moreover (3.15), supposed to be true for $n=k$, is also valid for $n=k+1$. In fact, we obtain

$$\beta_{k+1} \le \int_o^u \frac{\beta_k(\tau_1,\xi)}{1+\lambda(\tau_1+\xi)}\,d\tau_1 \le f_0 \int_o^u \frac{\tau_1^{\,k}}{k!}\,d\tau_1 = f_0 \frac{u^{k+1}}{(k+1)!}$$

Because of (3.15), the series $\sum_{n=1}^{\infty} \beta_n(u, x)$ for each fixed x is absolutely and uniformly convergent in every limited interval $I \subset R_0^+$ and therefore the succession $\{f_n(u,\xi)\}_{n\in N}$ converges uniformly to a function $f(u,\xi)$.

Considering now in (3.5) the limit for $n \to \infty$, one obtains

$$f(u,\xi) = f(0) + \int_0^u \frac{f(\tau)}{1+\lambda(\tau+\xi)}\,d\tau - \int_0^u \frac{1}{1+\lambda(\tau+\xi)} \cdot \int_o^\tau f(\tau-z,\xi)\,p(z)\,dz\,d\tau \qquad (3.17)$$

Then the existence of the solution of (3.1) (Davidson's problem) is proved.

About the uniqueness of the solution, see [5].

4. On the representation of the solution to Davidson's problem.

A representation's formula for the function $f(u,x)$ of problem (3.1) and therefore for the asymptotic probability of non-ruin will now be evaluated.

Substituting (3.10) into (3.8) we obtain:

$$\frac{\partial}{\partial u} f(u,\xi) = f_0\Big[1 + \int_0^u \frac{1}{1+\lambda(t_1+\xi)} \Big\{1 - P(t_1) +$$

$$+ \sum_{k=2}^{n} \int_0^{t_1} \cdots \int_0^{t_{k-1}} \prod_{i=2}^{k} \frac{1}{1+\lambda(t_i+\xi)} \big[1 - P(t_k)\big] \cdot$$

$$\cdot \prod_{j=1}^{k-1} \Big[1 - P(t_j - t_{j+1})\Big]\, dt_k \cdots dt_2 \Big\}\, dt_1 \Big] \qquad (4.1)$$

Setting

$$\alpha_1 = 1 - P(\tau_1) \qquad (4.2)$$

and, more generally,

$$\alpha_k = \int_0^{t_1} \cdots \int_o^{t_{k-1}} \big[1 - P(t_k)\big] \prod_{i=2}^{k} \frac{1}{1+\lambda(t_i+\xi)} \cdot$$

$$\cdot \prod_{j=1}^{k-1} \Big[1 - P(t_j - t_{j+1})\Big]\, dt_k \ldots\ldots dt_2 \qquad (k = 2,3,\ldots\ldots) \qquad (4.3)$$

we observe that, being

$$\alpha_k(t_1) \geq 0 \qquad \forall t_1 \geq 0 \qquad (4.4)$$

the series $\sum_k \alpha_k(\tau_1)$ results to be regular and, according to the theorem of monotonous convergence, calculating the limit under the sign of integral in (4.1) one obtains the solution

$$f(u,\xi) = f_0\left[1 + \int_0^u \frac{1}{1+\lambda(\tau_1+\xi)} \sum_{k=1}^{\infty} \gamma_k\, d\tau_1\right] \qquad (4.5)$$

The foregoing results can be summed up as follows.

Theorem: *Let $\lambda(z)$ be a continuous and positive function in $[0,+\infty]$ and $p(z)$ be a continuous and non-negative function in $[0,+\infty]$. Given*

$$P(z) = \int_0^z p(t)\, dt$$

the solution f of the problem (3.1) is provided by:

$$f(u,\xi) = f_0(\xi)\left[1+\int_0^u \frac{1}{1+\lambda(t_1+\xi)}\sum_{k=1}^{\infty}\alpha_k\, dt_1\right] \qquad (4.6)$$

where α_k *is given by (4.2) and (4.3).*

REFERENCES

1 ARFWEDSON G., *Some Problems in the Collective Theory of Risk*, Skand. AktuarTidskr, (1950).

2 BADOLATI E., *Sulla probabilità di rovina in ipotesi di caricamenti variabili*, Pubbl. n. 26, Istituto Statistica e Matematica, Ist. Univ. Navale, Napoli, (1987).

3 BADOLATI E., *Una nuova equazione integro-differenziale della teoria del rischio;* Pubbl. n. 28, Istituto Statistica e Matematica, Ist. Univ. Navale, Napoli, (1988).

4 DAVIDSON A., *On the Ruin Problem in the Collective Theory of Risk;* Skand. AktuarTidskr. , 3-4 Supplement, (1969).

5 DI LORENZO E., *Su un'equazione integro-differenziale della teoria collettiva del rischio in ipotesi di caricamenti variabili;* G.I.I.A, (1989).

6 DI LORENZO E., *Una formula di rappresentazione per la probabilità asintotica di non rovina;* Pubbl. Ist. Statistica, Univ. Salerno, (1990).

7 FAVATI L., GOSIO C., LISEI G., *Su una successione di approssimanti per difetto della probabilità di rovina nel caso di investimento della riserva;* Pubbl. Ist. Matem. Finanz., Univ. Genova, (1988).

8 FELLER W., *An introduction to Probability Theory; Wiley,* (1971).

9 GERBER H.U., *An Introduction to Mathematical Risk Theory,* Huebner Found., Irwin; Philadelphia, (1979).

10 GOSIO C., *Alcune limitazioni per la probabilità di rovina nel caso di caricamento non costante;* Pubbl. Ist. Matem. Finanz., Univ. Genova, (1989).

11 LAURIN I., *An Introduction into Lundberg's Theory of Risk;* Skand. AktuarTidskr., 1-2, (1930).

12 LISEI G., *Su un confine inferiore della probabilità di rovina nell'ambito della teoria collettiva del rischio;* G.I.I.A., (1983).

13 LISEI G., *Una successione di approssimanti per difetto della probabilità di rovina;* Pubbl. Ist. Matem. Finanz., Univ. Genova, (1988).

14 LUNDBERG F., *Forsakrangsteknisk riskutjamming.* I. Teori; Stockholm, (1926).

15 MARINA BORGHESANI M.E., *Alcune osservazioni sulla probabilità di rovina nel caso di caricamento non costante;* Pubbl. Ist. Matem. Finanz., Univ. Genova, (1988).

MODERATE-DENSITY CLOSE-CLOSED LOOP BURST ERROR DETECTING CODES

Bal Kishan Dass[1] Sapna Jain[2]

Abstract: In this paper, we study cyclic codes detecting a subclass of close-closed loop bursts viz. moderate-density close-closed loop bursts. A subclass of CT close-closed loop bursts called CT moderate-density close-closed loop bursts is also studied. A comparative study of the results obtained in this paper has also been made.

Keywords: Cyclic Codes, Moderate-Density Bursts, Close-Closed Loop Bursts, Error Detecting Codes.

1. Introduction

Burst errors are the most common type of errors that occur in several communication channels. Codes developed to detect and correct such errors have been studied extensively by many authors. The most successful early burst error correcting codes were due to Fire (1959). Fire in his report gave the idea of open and closed loop bursts defined as follows:

Definition 1. An **open loop burst** of length b is a vector all of whose non-zero components are confined to some b consecutive components, the first and the last of which is non-zero.

Definition 2. A **closed loop burst** of length b is a vector all of whose non-zero components are confined to some b consecutive components, the first and the last of which is non-zero and the number of positions from where the burst can start is n (i.e. it is possible to come back cyclically at the first position after the last position for enumeration of the length of the burst).

Definition 2 of closed loop burst can also be formulated mathematically on the lines Campopiano (1962) as follows:

Definition 2a. Let $V^n(q)$ be the set of all ordered n- tuples with components belonging to GF(q). Let X = $(a_0, a_1, ..., a_{n-1})$ be a vector in $V^n(q)$. Then X is

[1] Department of Mathematics, University of Delhi, Delhi 110 007, India
[2] Department of Mathematics, University of Delhi, Delhi 110 007, India

called a **closed loop burst of length b**, $2 \le b \le n$, if $\exists$ an i , $0 \le i \le n-1$, such that

$a_i.a_j \neq 0$ where j = (i + b - 1) modulo n

and $\begin{cases} a_{j+1} = a_{j+2} = ... = a_{i-1} = 0 \text{ if } i > j \\ a_0 = a_1 = ... = a_{i-1} = a_{j+1} = a_{j+2} = ... = a_{n-1} = 0 \text{ if } i < j \end{cases}$

There is yet another definition of a burst due to Chien and Tang (1965) which runs as follows:

Definition 3. A **CT burst of length b** is a vector all of whose non-zero components are confined to some b consecutive components, the first of which is non-zero.

Based on these definitions, Dass & Jain (2000) defined close-closed loop bursts, open-closed loop burst, CT close-closed loop burst, and CT open-closed loop burst and proved results for close-closed loop bursts and CT close-closed loop bursts. The definitions and the results proved by Dass & Jain (2000) are as follows:

Definition 4. Let $X = (a_0, a_1, ..., a_{n-1})$ be a vector in $V^n(q), a_i \in GF(q)$ and let $2 \le b \le n$. Then X is called a **close-closed loop burst of length b** if $\exists$ an i , $1 \le i \le b-1$ such that $a_{n-b+1}.a_{i-1} \neq 0, a_i = a_{i+1} = ... = a_{n-b+i-1} = 0.$

Definition 5. The class of open loop burst as considered in Definition 1 may be termed as **open-closed loop bursts**.

Definition 6. Let X= $(a_0, a_1, ..., a_{n-1})$ be a vector in $V^n(q)$ and $2 \le b \le n$. Then X is called a **CT close-closed loop bursts of length b** if $\exists$ an i , $1 \le i \le b-1$ such that $a_{n-b+i} \neq 0$; at least one of $a_0, a_1, ..., a_{i-1}$ is non-zero and $a_i = a_{i+1} = ... = a_{n-b+i-1} = 0$.

Definition 7. The class of CT open loop burst as considered in Definition 3 may be termed as **CT open-closed loop bursts**.

Theorem A. An (n, k) cyclic can not detect any close-closed loop burst of length b where $2 \le b \le k+1$.

Theorem B. The fraction of close-closed loop bursts of length b ($2 \le b \le k+1$) that goes undetected to the total number of close-closed loop bursts in any (n, k) cyclic code is

$$= \frac{q^{k-2b+3}(q^{b-1}-1)}{(b-1)(q-1)^2} .$$

Theorem C. An (n, k) cyclic code can not detect any CT close-closed loop burst of length b where $2 \le b \le k+1$.

Theorem D. The fraction of CT close-closed loop burst of length b $(2 \le b \le k+1)$ that goes undetected to the total number of CT close-closed loop bursts in any (n, k) cyclic code is

$$= \frac{q^{k-b+1}(q^{b-1}-1)}{q^{b-1}\big((b-1)q-b\big)+1} .$$

There are of course many situations in which errors occur in the form of bursts but not all digits inside the burst get corrupted. Usually, the weight of the burst lies between two numbers w_1 and w_2 such that $2 \le w_1 \le w_2 \le$ length of burst. Such bursts are known as **moderate-density bursts**. Moderate-density bursts with respect to close-closed loop burst are known as moderate-density close-closed loop bursts and are defined as follows:

Definition 8. A close-close loop burst of length b whose weight lies between w_1 and w_2, $2 \le w_1 \le w_2 \le b$, is called a **moderate-density close-closed loop burst**.

The development of codes which detect/correct moderate-density close-closed loop bursts can economize in the number of parity check digits required, suitably reducing the redundancy of the code or in the other words, suitably increasing the efficiency of transmission. In the second section of this paper, we obtain results similar to Theorem A and B for moderate-density close-closed loop bursts whereas in the third section, we obtain results similar to Theorems C and D for CT moderate-density close-closed loop bursts. The last section viz. Section 4 gives a comparison of the results obtained in Section 2 and Section 3.

In what follows, an (n, k) cyclic code over GF(q) is taken as an ideal in the algebra of polynomials modulo the polynomial $X^n - 1$.

2. Moderate-Density Close-Closed Loop Burst Error Detection

In this section, we obtain results of Theorems A and B for moderate- density close-closed loop bursts.

Theorem 1. An (n, k) cyclic codes can not detect any moderate-density close-closed loop burst of length b with weight lying between w_1 and w_2 $(w_1 \le w_2 \le b)$ where $2 \le b \le k+1$.

Proof. There is no deviation in the final conclusion of this theorem from that of Theorem A because the proof is based on the length of the burst giving rise to a polynomial which is of the same degree even when the weight consideration over the burst is considered. Hence the proof is omitted.

Q.E.D.

Theorem 2. The fraction of moderate-density close-closed loop bursts of length b $(2 \le b \le k+1)$ with weight lying between w_1 and w_2 that goes undetected to the total number of moderate-density close-closed loop bursts in any (n, k) cyclic code is

$$= \frac{q^k (1-q^{-b+1})}{\sum_{i=1}^{b-1} \left\{ \sum_{r_1=1}^{w_2-1} \left\{ \binom{b-i-1}{r_1-1} (q-1)^{r_1+1} \sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1} \binom{i-1}{r_2-1} (q-1)^{r_2-1} \right\} \right\}}$$

where $\langle w_1 - r_1, 1 \rangle = \max.\{w_1 - r_1, 1\}$

Proof. Let r(X) denote a moderate-density close-closed loop burst of length $b(2 \le b \le k+1)$ with weight w lying between w_1 and w_2 $(w_1 \le w_2 \le b)$. Let g(X) denote the generator polynomial of the code of degree n – k.

Now r (X) will be of the form

$$r(X) = X^{n-b+i}(a_{n-b+i} + a_{n-b+i+1}X + \ldots + a_{n-1}X^{b-1-i}) + (a_0 + a_1X + a_2X^2 + \ldots + a_{i-1}X^{i-1});\ 1 \le i \le b-1, a_{n-b+i}, a_{i-1} \ne 0$$

and the number of non-zero coefficients, including a_{n-b+i}, a_{i-1} lies between w_1 and w_2.

$$= X^{n-b+i} r_1(X) + r_2(X), \text{ say}$$

where $r_1(X) = a_{n-b+i} + a_{n-b+i+i}X + \ldots + a_{n-1}X^{b-1-i}$

and $r_2(X) = a_0 + a_1X + a_2X^2 + \ldots + a_{i-1}X^{i-1}$.

Let r_1 be the number of non-zero coefficients in $r_1(X)$ and r_2 be the number of non-zero coefficients in $r_2(X)$,

where

$$1 \le r_1 \le w_2 - 1$$

and $1 \le r_2 \le w_2 - 1$

Such that $w_1 \le r_1 + r_2 \le w_2$.

For any fixed value of i, let us give different values of r_1.

(i) Let $r_1 = 1$. Then $\langle w_1 - 1,\ 1\rangle \le r_2 \le w_2 - 1$

($\because\ w_1 \le r_1 + r_2 \le w_2 \Rightarrow w_1 - r_1 \le r_2 \le w_2 - r_1$, also $r_2 \ge 1$

$\therefore \langle w_1 - r_1, 1\rangle \le r_2 \le w_2 - r_1$)

where $\langle w_1 - r_1, 1\rangle = \max\{w_1 - r_1, 1\}$.

We have then

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{0}(q-1)^0$

Number of polynomials of type $r_2(X) = \sum_{r_2=\langle w_1-1,1\rangle}^{w_2-1} (q-1)\binom{i-1}{r_2-1}(q-1)^{r_2-1}$

$\therefore$ Number of polynomials of type r(X)

$$= \binom{b-i-1}{0}(q-1)^2 \sum_{r_2=\langle w_1-1,1\rangle}^{w_2-1} \binom{i-1}{r_2-1}(q-1)^{r_2-1}$$

(ii) For $r_1 = 2$ we get $\langle w_1 - 2, 1\rangle \le r_2 \le w_2 - 2$

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{1}(q-1)$

Number of polynomials of type $r_2(X) = \sum_{r_2=\langle w_1-2,1\rangle}^{w_2-2} (q-1)\binom{i-1}{r_2-1}(q-1)^{r_2-1}$

$\therefore$ Number of polynomials of type r(X)

$$= \binom{b-i-1}{1}(q-1)^3 \sum_{r_2=\langle w_1-2,1\rangle}^{w_2-2} \binom{i-1}{r_2-1}(q-1)^{r_2-1}$$

Continuing the computation for various values of $r_1 = 3,4,...,$ we finally, have

$r_1 = w_2 - 1 \Rightarrow r_2 = 1$

and

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{w_2-2}(q-1)^{w_2-2}$

Number of polynomials of type $r_2(X) = = \sum_{r_2=1}^{1} (q-1)\binom{i-1}{r_2-1}(q-1)^{r_2-1}$

$\therefore$ Number of polynomials of type r(X)

$$= \binom{b-i-1}{w_2-2}(q-1)^{w_2} \sum_{r_2=1}^{1} \binom{i-1}{r_2-1}(q-1)^{r_2-1}$$

So, for a fixed value of i,

Number of polynomials of type r(X)

$$= \sum_{r_1=1}^{w_2-1} \left\{ \binom{b-i-1}{r_1-1} (q-1)^{r_1+1} \sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1} \binom{i-1}{r_2-1} (q-1)^{r_2-1} \right\}$$

Summing over i, we get

Total number of polynomials of type r(X)

$$= \sum_{i=1}^{b-1} \left\{ \sum_{r_1=1}^{w_2-1} \left\{ \binom{b-i-1}{r_1-1} (q-1)^{r_1+1} \sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1} \binom{i-1}{r_2-1} (q-1)^{r_2-1} \right\} \right\}$$

Again, r(X) will go undetected if g(X) divides r(X)

$\Rightarrow$ r(X) =g(X)Q(X) for some polynomials Q(X)

$\Rightarrow \quad X^{n-b+1} r_1(X) + r_2(X) = \text{g(X)Q(X)}$

Now, number of polynomials of type Q(X) = $q^k (1-q^{-b+1})$ (refer[3])

$\therefore$ Ratio of moderate-density close-closed loop bursts that goes undetected to the total number of moderate-density close-closed loop bursts is

$$= \frac{q^k (1-q^{-b+1})}{\sum_{i=1}^{b-1} \left\{ \sum_{r_1=1}^{w_2-1} \left\{ \binom{b-i-1}{r_1-1} (q-1)^{r_1+1} \sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1} \binom{i-1}{r_2-1} (q-1)^{r_2-1} \right\} \right\}}$$

where $\langle w_1 - r_1, 1 \rangle = \max.\{w_1 - r_1, 1\}$

Hence the proof. **Q.E.D.**

Special Cases. (i) For $b = w_1 = w_2 = 2$, the ratio obtained in the preceding theorem reduces to the ratio given in Theorem B for b=2 and the ratio in each case becomes

$$\frac{q^{k-1}}{(q-1)}.$$

(ii) For $w_1 = 2$, the result obtained in the preceding theorem reduces to the case of low-density close-closed loop bursts considered by Dass & Jain (2000).

(iii) For $w_2 = b$, the result obtained in the preceding theorem reduces to the case for high-density close-closed loop bursts, considered by Dass & Jain (2000).

3. CT Moderate-Density Close-Closed Loop Burst Error Detection

In this section we extend the studies made in Section 2 for CT moderate-density close-closed loop bursts. Firstly, we obtain the following result, the proof of which is omitted.

Theorem 3. An (n, k) cyclic code can not detect any CT moderate-density close-closed loop burst of length $b\,(2 \le b \le k+1)$ with weight lying between w_1 and $w_2\,(w_1 \le w_2 \le b)$.

We now prove the following result.

Theorem 4. The fraction of CT moderate-density close-closed loop bursts of length b $(2 \le b \le k+1)$ with weight lying between w_1 and w_2 that goes undetected to the total number of CT moderate-density close-closed loop bursts in any (n, k) cyclic code is

$$= \frac{q^k (1-q^{-b+1})}{\sum_{i=1}^{b-1}\left\{\sum_{r_1=1}^{w_2-1}\left\{\binom{b-i-1}{r_1-1}(q-1)^{r_1}\sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1}\binom{i}{r_2}(q-1)^{r_2}\right\}\right\}}$$

where $\langle w_1 - r_1, 1\rangle = \max.\{w_1 - r_1, 1\}$

Proof. Let r(X) denote a CT moderate-density close-closed loop burst of length b $(2 \le b \le k+1)$ with weight lying between w_1 and w_2 $(w_1 \le w_2 \le b)$. Let g(X) denote the generator polynomial of the code of degree n – k .
Now r(X) will be of the form

r(X) $= X^{n-b+i}(a_{n-b+i} + a_{n-b+i+1}X + ... + a_{n-1}X^{b-1-i})$

$+ (a_0 + a_1X + a_2X^2 + ... + a_{i-1}X^{i-1})$; $i \le 1 \le b-1, a_{n-b+i} \ne 0$ and the number

of non-zero coefficients, including a_{n-b+i}, a_{i-1} lies between w_1 and w_2 .

$= X^{n-b+i} r_1(X) + r_2(X)$, say

where $r_1(X) = a_{n-b+i} + a_{n-b+i+i}X + ... + a_{n-1}X^{b-1-i}$

and $r_2(X) = a_0 + a_1X + a_2X^2 + ... + a_{i-1}X^{i-1}$.

Let r_1 be the number of non-zero coefficients in $r_1(X)$ and r_2 be the number of non-zero coefficients in $r_2(X)$,

where

$$1 \le r_1 \le w_2 - 1$$

and $$1 \le r_2 \le w_2 - 1$$

Such that $w_1 \le r_1 + r_2 \le w_2$.

For any fixed value of i, let us give different values of r_1.

(i) Let $r_1 = 1$. Then $\langle w_1 - 1, 1\rangle \le r_2 \le w_2 - 1$ and

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{0}(q-1)^0$

Number of polynomials of type $r_2(X) = \sum_{r_2=\langle w_1-1,1\rangle}^{w_2-1} \binom{i}{r_2}(q-1)^{r_2}$

$\therefore$ Number of polynomials of type r(X)

$= \binom{b-i-1}{0}(q-1)\sum_{r_2=\langle w_1-1,1\rangle}^{w_2-1}\binom{i}{r_2}(q-1)^{r_2}$

(ii) Let $r_1 = 2$ we get $\langle w_1 - 2, 1\rangle \le r_2 \le w_2 - 2$

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{1}(q-1)$

Number of polynomials of type $r_2(X) = \sum_{r_2=\langle w_1-2,1\rangle}^{w_2-2}\binom{i}{r_2}(q-1)^{r_2}$

$\therefore$ Number of polynomials of type r(X)

$= \binom{b-i-1}{1}(q-1)^2\sum_{r_2=\langle w_1-2,1\rangle}^{w_2-2}\binom{i}{r_2}(q-1)^{r_2}$

Continuing the computation for various values of $r_1 = 3,4,...,$ we finally, have

$r_1 = w_2 - 1 \Rightarrow r_2 = 1$ and

Number of polynomials of type $r_1(X) = (q-1)\binom{b-i-1}{w_2-2}(q-1)^{w_2-2}$

Number of polynomials of type $r_2(X) = \sum_{r_2=1}^{1}\binom{i}{r_2}(q-1)^{r_2}$

Number of polynomials of type r(X)

$= \binom{b-i-1}{w_2-2}(q-1)^{w_2-1}\sum_{r_2=1}^{1}\binom{i}{r_2}(q-1)^{r_2}$

So, for a fixed value of i,

Number of polynomials of type r(X)

$= \sum_{r_1=1}^{w_2-1}\left\{\binom{b-i-1}{r_1-1}(q-1)^{r_1}\sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1}\binom{i}{r_2}(q-1)^{r_2}\right\}$

Summing over i , we get

Total number of polynomials of type r(X)

$= \sum_{i=1}^{b-1}\left\{\sum_{r_1=1}^{w_2-1}\left\{\binom{b-i-1}{r_1-1}(q-1)^{r_1}\sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1}\binom{i}{r_2}(q-1)^{r_2}\right\}\right.$

Again, r(X) will go undetected if g(X) divides r(X)

$\Rightarrow$ r(X) = g(X)Q(X) for some polynomials Q(X)

$\Rightarrow \; X^{n-b+i} r_1(X) + r_2(X) = g(X)Q(X)$

Now, number of polynomials of type Q(X) = $q^k (1 - q^{-b+1})$ (refer[3])

$\therefore$ Ratio of moderate-density close-closed loop bursts that goes undetected to the total number of moderate-density close-closed loop bursts is

$$= \frac{q^k (1 - q^{-b+1})}{\sum_{i=1}^{b-1} \left\{ \sum_{r_1=1}^{w_2-1} \left\{ \binom{b-i-1}{r_1-1} (q-1)^{r_1} \sum_{r_2=\langle w_1-r_1,1\rangle}^{w_2-r_1} \binom{i}{r_2} (q-1)^{r_2} \right\} \right.}$$

where $\langle w_1 - r_1, 1 \rangle = \max.\{w_1 - r_1, 1\}$

Hence the proof.

Q.E.D.

Special Cases. (i) For $b = w_1 = w_2 = 2,$ the ratio obtained in the preceding theorem reduces to the ratio given in Theorem B for b=2 and the ratio in each case becomes

$$\frac{q^{k-1}}{(q-1)}.$$

(ii) For $w_1 = 2,$ the result obtained in the preceding theorem reduces to the case of low-density close-closed loop bursts considered by Dass & Jain (2000).

(iii) For $w_2 = b,$ the result obtained in the preceding theorem reduces to the case for high-density close-closed loop bursts, considered by Dass & Jain (2000).

4. Comparative Study

In this section, we present the comparison of the results obtained in Section 2 and Section 3 viz. Theorem 2 and Theorem 4. The comparison has been presented in the form of a table by taking specific values of b, w_1 and w_2 in the binary case. For $b = w_1 = w_2 = 2$, both definitions viz. of moderate-density close-closed loop burst and of CT moderate-density close-closed loop burst coincide. Therefore, we start comparing the results for b=3, and onwards.

TABLE $[q = 2]$

Moderate-Density Close-Closed Loop Bursts (Theorem 2)		CT Moderate-Density Close-Closed Loop Bursts (Theorem 4)
		$[b = 3; w_1 = 2, w_2 = 2]$
$k = 2$	1.50	1.00
$k = 3$	3.00	2.00
$k = 4$	6.00	4.00
		$[b = 3; w_1 = 2, w_2 = 3]$
$k = 2$	0.75	0.60
$k = 3$	1.50	1.20
$k = 4$	3.00	2.40
		$[b = 3; w_1 = 3, w_2 = 3]$
$k = 2$	1.50	1.50
$k = 3$	3.00	3.00
$k = 4$	6.00	6.00
		$[b = 4; w_1 = 2, w_2 = 2]$
$k = 3$	2.33	1.16
$k = 4$	4.66	2.33
$k = 5$	9.33	4.66
		$[b = 4; w_1 = 2, w_2 = 3]$
$k = 3$	0.77	0.50
$k = 4$	1.55	1.00
$k = 5$	3.11	2.00
		$[b = 4; w_1 = 2, w_2 = 4]$
$k = 3$	0.58	0.41
$k = 4$	1.66	0.82
$k = 5$	2.33	1.64
		$[b = 4; w_1 = 3, w_2 = 3]$
$k = 3$	1.16	0.87
$k = 4$	2.33	1.75
$k = 5$	4.66	3.50

		$[b=4; w_1=3, w_2=4]$
$k=3$	0.77	0.63
$k=4$	1.55	1.27
$k=5$	3.11	2.54
		$[b=4; w_1=4, w_2=4]$
$k=3$	2.33	2.33
$k=4$	4.66	4.66
$k=5$	9.33	9.33

		$[b=5; w_1=2, w_2=2]$
$k=4$	3.75	1.50
$k=5$	7.50	3.00
$k=6$	15.00	6.00
		$[b=5; w_1=2, w_2=3]$
$k=4$	0.93	0.50
$k=5$	1.87	1.00
$k=6$	3.75	2.00
		$[b=5; w_1=2, w_2=4]$
$k=4$	0.53	0.33
$k=5$	1.07	0.66
$k=6$	2.14	1.33
		$[b=5; w_1=2, w_2=5]$
$k=4$	0.46	0.30
$k=5$	0.93	0.61
$k=6$	1.87	1.22
		$[b=5; w_1=3, w_2=3]$
$k=4$	1.25	0.75
$k=5$	2.50	1.50
$k=6$	5.00	3.00
		$[b=5; w_1=3, w_2=4]$
$k=4$	0.62	0.42
$k=5$	1.25	0.85
$k=6$	2.50	1.71

		$[b=5; w_1=3, w_2=5]$
$k=4$	0.53	0.38
$k=5$	1.07	0.76
$k=6$	2.14	1.53
		$[b=5; w_1=4, w_2=4]$
$k=4$	1.25	1.00
$k=5$	2.50	2.00
$k=6$	5.00	4.00
		$[b=5; w_1=4, w_2=5]$
$k=4$	0.93	0.78
$k=5$	1.87	1.57
$k=6$	3.75	3.15
		$[b=5; w_1=5, w_2=5]$
$k=4$	3.75	3.75
$k=5$	7.50	7.50
$k=6$	15.00	15.00

Note. The fractions have been calculated up to 2 decimal places.

Acknowledgement. The second author wishes to thank *University Grants Commission* for providing grant (vide Ref. No. F-13-3/99(SR-I)) under *Minor Research Project* to carry out this research work.

References

1. C.N. Campopiano (1962), Bounds on Burst Error Correcting Codes, **IRE Trans.**, IT-8, pp. 257-259.

2. R.T. Chien and D.T. Tang (1965), On Definitions of a Burst **IBM J. Res. & Devlop.** , July pp. 292-293.

3. B.K. Dass and Sapna Jain (2000), On a Class of Closed Loop Bursts Error Detecting Codes, **International Journal of Nonlinear Sciences and Numerical Simulation** 2, pp. 305-306, 2001.

4. B.K. Dass and Sapna Jain (2000), Low-Density Close-Closed Loop Burst Error Detecting Codes, accepted for publication in **Korean Journal of Computational and Applied Mathematics**.

5. B.K. Dass and Sapna Jain (2000), High-Density Close-Closed Loop Burst Error Detecting Codes, submitted.

6. P. Fire (1959), A Class of Multiple-Error-Correcting Binary Codes for Non-Independent Errors, Sylvania Report RSL-E-2, Sylavania Recon. Sys. Lab., Mountain View, California.

7. W.W. Peterson (1961), **Error Correcting Codes**, Cambridge, Mass: The M.I.T. Press.

DYNAMIC PROGRAMMING APPROACH TO TESTING RESOURCE ALLOCATION PROBLEM FOR MODULAR SOFTWARE

P.K. Kapur[1] P.C. Jha[1] A.K. Bardhan[1]

Abstract
Testing phase of a software begins with module testing. During this period modules are tested independently to remove maximum possible number of faults within a specified time limit or testing resource budget. This gives rise to some interesting optimization problems, which are discussed in this paper. Two Optimization models are proposed for optimal allocation of testing resources among the modules of a Software. In the first model, we maximize the total fault removal, subject to budgetary Constraint. In the second model, additional constraint representing aspiration level for fault removals for each module of the software is added. These models are solved using dynamic programming technique. The methods have been illustrated through numerical examples.

Key words: Software Reliability, Non Homogeneous Poisson Process, Resource Allocation, Dynamic Programming

1. Introduction

Growth in software engineering technology has led to production of software for highly complex situations occurring in industry, scientific research, defense and day to day life. Consequently, the dependence of mankind on computers and computer-based systems is increasing day by day. Any failure in these systems can cost heavily in terms of money and/or human lives. Though high reliability of hardware part of these systems can be guaranteed, the same cannot be said for software. Therefore a lot of importance is attached to the testing phase of the software development process, where around half the developmental resources are used [8]. Essentially testing is a process of executing a program with the explicit intention of finding faults and it is this phase, which is amendable to mathematical modeling.

It is always desirable to remove a substantial number of faults from the software. In fact the reliability of a software is directly proportional to the number of faults removed. Hence the problem of maximization of software reliability is identical to

[1] Department of Operational Research, Faculty of Mathematical Sciences, University of Delhi, Delhi 110007, INDIA

that of maximization of fault removal. At the same time testing resource are not unlimited, and they need to be judiciously used. In this paper we discuss and solve such a management problem of allocation of testing resources among modules, through a Software Reliability Growth Model (SRGM). A Software Reliability Growth Model (SRGM) is a relationship between the number of faults removed from a software and the execution time/CPU time/calendar time. Several attempts have been made to represent the actual testing environment through SRGMs [1,4,5,9]. These models have been used to predict the fault content, reliability and release time of a software. SRGMs have also been used to manage the testing phase. Again large software consists of modules. Often these modules are developed independently and each module may contain different number of faults and that of different severity. Therefore distinct SRGMs should be used to represent the testing process of each module, as testing for these modules are done independently. An SRGM with testing effort [9] has been chosen to represent the fault removal process for the two optimization problems discussed in this paper.

The first optimization model (P1) maximizes the total number of faults expected to be removed, when available testing resource is known. The management normally aspires for some reliability level that can be translated in terms of number of faults removed. In the second optimization model (P2) we add a constraint in (P1) in terms of minimum number of faults aspired to be removed from each module. Dynamic programming technique is used to solve these problems. This is the first time that this has been done in software engineering, according to our knowledge. Dynamic programming approach, which is easy to solve and understand provides global optima for these problems. The methodology discussed in the paper has been illustrated through numerical examples.

Notations

N	:	Number of modules in the Software (>1)
a_i	:	Expected number of faults in the i^{th} module (i=1,2,…,N)
b_i	:	Proportionality constant for the i^{th} module
$x_i(t)$	:	Current testing effort expenditure at testing time t and $X_i(t) = \int_0^t x_i(w)dw$ for i^{th} module
X_i, Z	:	The amount of testing resource to be allocated to the i^{th} module and total testing resource available.
$m_i(t)$	:	Number of faults removed in (0,t] the ith module, mean value function of NHPP, i = 1,…,N
T	:	Total testing time
X_i^*	:	Optimal value of X_i , i = 1,…,N
$f_n(Z)$	:	Optimal number of faults removed upto nth modules (i.e. corresponding to nth stage in a Dynamic Programming algorithm)

a_{io} : Aspiration level of ith module (i.e. number of faults desired to be removed from i^{th} module)

p_i : The minimum proportion of total faults to be removed from i^{th} module.

2. Mathematical Modelling

2.1 Resource Allocation Problem

Consider a software having N modules, which are being tested independently for removing faults lying dormant in them. The duration of module testing is often fixed when scheduling is done for the whole testing phase. Hence limited resources are available, that need to be allocated judiciously. If m_i faults are expected to be removed from the i^{th} module with effort X_i, the resulting testing resource allocation problem can be stated as follows [5,6].

$$\max \quad \sum_{i=1}^{N} m_i$$

subject to

$$\sum_{i=1}^{N} X_i = Z \,,\; X_i \geq 0 \,,\; i = 1,\ldots,N \qquad \ldots \qquad \ldots \qquad \text{(P1)}$$

Above optimization problem is the simplest one as it considers the resource constraint only. Later in this paper, we incorporate additional constraints to the basic model. For solving (P1) a functional relationship between fault removal and resource consumption is required, which is discussed in the following section.

2.2 SRGM For Modules

A Software Reliability Growth Model explains the time dependent behavior of fault removal. As modules are tested independently distinct SRGMs would represent their reliability growth. The influence of testing effort can also be included in the SRGMs [9]. In this paper we discuss the resource allocation problem using such a SRGM for the modules.

Model Assumptions

1. Software consist of a finite number of modules and testing for each module is done independently

2. A module is subject to failures at random time caused by faults remaining in the software.
3. On a failure, the fault causing that failure is immediately removed and no new faults are introduced.
4. Fault removal phenomenon is modelled by Non Homogeneous Poisson Process (NHPP).
5. The expected number of faults removed in $(t, t+\Delta t)$ to the current testing resource is proportional to the expected remaining number of faults.

Under assumption 5, following differential equation may easily be written for i^{th} module

$$\frac{\frac{d}{dt} m_i(t)}{x_i(t)} = b_i(a_i - m_i(t)), \; i = 1,\ldots,N \qquad \ldots \quad \ldots \quad (1)$$

Solving equation (1) with the initial condition that, at $t = 0$, $X_i(t) = 0$, $m_i(t) = 0$ we get

$$m_i(t) = a_i(1 - e^{-b_i X_i(t)}), \; i = 1,\ldots,N \qquad \ldots \quad \ldots \quad (2)$$

To describe the behaviour of testing effort, either Exponential or Rayleigh function has been used [5,9]. Both can be derived form the assumption that, " the testing effort rate is proportional to the testing resource available".

$$\frac{dX_i(t)}{dt} = c_i(t)[\alpha_i - X_i(t)], \; i = 1,\ldots,N \qquad \ldots \quad \ldots \quad (3)$$

where $c_i(t)$ is the time dependent rate at which testing resources are consumed, with respect to the remaining available resources. Solving equation (3) under the initial condition $X(0) = 0$ we get

$$X_i(t) = \alpha_i \left[1 - \exp \int_0^t c_i(k)dk \right], \; i = 1,\ldots, N \qquad \ldots \quad \ldots \quad (4)$$

When $c(t) = \beta$, a constant

$$X_i(t) = \alpha_i(1 - e^{-\beta_i t}), \; i = 1,\ldots,N \qquad \ldots \quad \ldots \quad (5)$$

If $c(t) = \beta.t$, (1) gives a Rayleigh type curve

$$X_i(t) = \alpha_i(1 - e^{-\beta_i \frac{t^2}{2}}), \; i = 1,\ldots,N \qquad \ldots \quad \ldots \quad (6)$$

In this paper we have chosen exponential function (5) to represent testing effort in the optimization problems.

2.3 Estimation Of Parameters

The testing effort data are given in the form of testing effort $x_k (x_1 < x_2 < ... < x_n)$ consumed in time $(0, t_i]$; $i = 1,2,..,n$. The testing effort model parameters α_i and β_i can be estimated by the method of least squares as follows.

Minimize $\sum_{i=1}^{n} \left[X_i - \hat{X} \right]$

subject to $\hat{X}_n = X_n$ (i.e. the estimated value of testing effort is equal to the actual value).

Once the estimates of α_i and β_i are known, the parameters of the SRGMs (2) for the modules can be estimated through Maximum Likelihood Estimation method using the underlying Stochastic Process, which is described by a Non Homogeneous Poisson Process. During estimation, estimated values of α_i and β_i are kept fixed. If the fault removal data for a module is given in the form of cumulative number of faults removed y_j in time $(0,t_j]$. The likelihood function for that module is given as

$$L_i(a_i, b_i / (y_i, W_i) = \prod_{j=1}^{n} \frac{\left[m_i(t_j) - m_i(t_{j-1}) \right]^{y_j - y_{j-1}}}{(y_j - y_{j-1})!} e^{-(m_i(t_j) - m_i(t_{j-1}))}$$

3. Optimal Allocation Of Resources

From the estimates of parameters of SRGMs for modules, the total fault content in the software $\sum_{i=1}^{N} a_i$ is known. Modules testing aims at removing maximum number of them, within available resources. Hence (P1) can be restated as

Maximize $\sum_{i=1}^{N} m_i(X_i) = \sum_{i=1}^{N} a_i (1 - e^{-b_i X_i})$

Subject to $\sum_{i=1}^{N} X_i \le Z$, $X_i \ge 0$ $\quad i = 1, \ldots, N \quad \ldots \quad$ (P1A)

(P1A) can be solved using Dynamic Programming Approach. From Bellman's principle of optimality, we can write the following recursive equation [2].

$$f_1(Z) = \max_{X_1 = Z_1} \left\{ a_1 (1 - e^{-b_1 X_1}) \right\}$$

$$f_n(Z) = \max_{0 \le X_n \le Z} \left\{ a_n (1 - e^{-b_n X_n}) + f_{n-1}(Z - X_n) \right\}, \; n = 2,\ldots,N \; \ldots \qquad (7)$$

To index the modules, they can be arranged in descending order of their values of $a_i b_i$ *i.e.* $a_1 b_1 \ge a_2 b_2 \ge \ldots \ge a_N b_N$. Through this approach resources are allocated to the modules sequentially. But for some values of Z ($Z < Z_r$) one or more modules with higher index number i.e. having lower detectability may not get any allocation. We summarize this result in the following simple theorem.

Theorem - 1

If for any $n = 2,\ldots,N$; $1 \le e^{-\mu_{n-1} Z} \le \dfrac{\mu_{n-1} V_{n-1}}{a_n b_n}$, then values of $X_n, X_{n+1}, \ldots X_N$ are zero and problem reduces to (n-1) stage problem with

$$X_r = \frac{1}{b_r + \mu_{r-1}} \left[\mu_{r-1} Z - \log \left(\frac{\mu_{r-1} V_{r-1}}{a_r b_r} \right) \right], \; r = 1,\ldots,(n\text{-}1) \qquad \ldots \qquad (8)$$

where $\mu_i = \dfrac{1}{\sum_{j=1}^{i} \left(\frac{1}{b_j}\right)}$ and $\mu_i V_i = \prod_{j=1}^{i} (a_j b_j)^{(\mu_i / b_j)}$, $i = 1,\ldots,N$

Proof of the theorem is given in appendix.

As a result of the above allocation procedure, some modules may not be tested at all. This situation is not advisable. Again management often aspires to achieve certain minimum reliability level for the software and that for each module of the Software i.e. a certain percentages of the fault content in each module of the Software is desired to be removed. Hence (P1) needs to be suitably modified to maximize removal of faults in the software under resource constraint and minimum desired level of faults to be removed from each of the modules in the software. The resulting testing resource allocation problem can be stated as follows:

$$\max \sum_{i=1}^{N} m_i = \sum_{i=1}^{N} a_i (1 - e^{-b_i X_i})$$

subject to

$$m_i = a_i (1 - e^{-b_i X_i}) \ge p_i a_i = a_{i_0} \; , \; i = 1, \ldots, N$$

$$\sum_{i=1}^{N} X_i \le Z\ ,\ \ X_i \ge 0\ ,\ i = 1, \ldots, N \qquad \ldots \qquad \text{(P2)}$$

(P2) can be solved using Dynamic Programming Approach either by reducing the dimensionality of the problem through Lagrange multiplier or converting to (P1) by substitution. We first consider the dimensionality reduction in Dynamic Programming Approach [2] as follows.

$$\max_{X} \min_{\alpha} \phi(X, \alpha) = \sum_{i=1}^{N} \left[a_i (1 - e^{-b_i X_i}) + \alpha_i \left\{ a_i (1 - e^{-b_i X_i}) - a_{i_0} \right\} \right]$$

$$\text{subject to } \sum_{i=1}^{N} X_i \le Z \qquad X_i, \alpha_i \ge 0 \ \ i = 1, \ldots, N \qquad \text{(P3)}$$

Where α_i (i = 1, … , N) is Lagrange multiplier for i^{th} constraint corresponding to the i^{th} module. The above problem can be solved by Dynamic Programming approach in which Kuhn-Tuckker optimality conditions are obtained at each stage [2]. At any stage α_i (i = 1,…,N) can be zero or non-zero depending upon ineffectiveness or effectiveness of constraint respectively. Hence each stage has two possibilities and corresponding to each possibility of preceding stage present stage has two possibilities. So at any stage i, total number of cases is 2^{i-1}. Infact, above problem reduces to that of finding an optimal path by searching for an optimal solution at each stage in which only one option could be chosen. This procedure does not provide a closed form solution. Hence without further elaboration of the above method, the substitution method is adopted for converting the problem (P2) to the problem (P1) as follows:

$m_i(X_i) \ge a_{i_0}$ implies $a_{\text{i}}(1 - e^{-b_{\text{i}} X_{\text{i}}}) \ge a_{i_0}$

Hence, $X_i \ge -\frac{1}{b_{\text{i}}} \log\left[1 - \frac{a_{\text{i}_0}}{a_{\text{i}}} \right] = Z_i$ (say), i = 1, … , N

Therefore (P2) can be restated as,

Maximize $\sum_{i=1}^{N} m_i = \sum_{i=1}^{N} a_i (1 - e^{-b_i X_i})$

subject to $X_i \ge Z_i \quad i = 1, \ldots, N$

$$\sum_{i=1}^{N} X_i \le Z\ ,\ X_i \ge 0\ ,\ i = 1, \ldots, N \qquad \text{(P4)}$$

Let $Y_i = X_i - Z_i$ (i = 1, … , N), then (P4) can be written as the problem (P1) given below

$$\max \sum_{i=1}^{N} m_i = \max \sum_{i=1}^{N} \overline{a}_i (1 - e^{-b_i Y_i})$$

subject to

$$\sum_{i=1}^{N} Y_i \le Z - \sum_{i=1}^{N} Z_i = \overline{Z} \text{ (say)}$$

$$Y_i \ge 0, \; i = 1, \ldots, N$$

$$\overline{a_i} = a_i - a_{i_0}, \; i = 1, \ldots, N \qquad \text{(P5)}$$

The Problem (P5) is similar to the problem (P1) and hence using theorem-1 the problem (P5) can also be solved.

If for any i = 2, … , N $1 \le e^{-\mu_i \overline{Z}} \le \frac{\mu_{i-1} V_{i-1}}{\overline{a_i} b_i}$,then $Y_i, Y_{i+1}, \ldots, Y_N$ are zeroes, then problem (P5) reduces to a $(i-1)$ stage problem and its solution is given as

$$Y_n = \frac{1}{b_n + \mu_{n+1}} \left[\mu_{n-1} Z - \log \left(\frac{\overline{\mu}_{n-1} V_{n-1}}{\overline{a_n} b_n} \right) \right], \; n = 1, \ldots, (i\text{-}1) \qquad \ldots \qquad (9)$$

$$f(\overline{Z}) = \sum_{n=1}^{i-1} \overline{a}_i - V_n e^{-\mu_n \overline{Z}} \qquad \ldots \qquad \ldots \qquad (10)$$

Through equation (9) optimal allocation of resources to the modules can be calculated. In the following section we numerically illustrate these results.

4. Numerical Example

It is assumed that parameters a_i and b_i for the i[th] module (i=1,.....N) are already estimated using the software failure data. Consider a software having 10 modules whose parameter estimates are as given in Table-1. Suppose the total resource available for testing is 97000. First the problem (P1) is solved and from the recursion equation (7) optimal allocation of resources (X_i^*) for the modules are computed. These are listed in Table-1 along with the corresponding expected number of fault removed, percentages of faults removed and faults remaining for each module. The total number of faults that can be removed through this allocation is 152 (i.e. 60.6% of the fault content is removed from the Software). It is observed

that in some modules (module-9,10) the remaining faults after allocation is high. This can lead to frequent failure during operational phase. Obviously this will not satisfy the developer and he may desire that at least 50% of fault content from each of the modules of the Software is removed (i.e. p_i=0.5 for each i = 1…10). Since faults in each module are integral values, nearest integer larger then 50% of the fault content in each module is taken as lower limit that has to be removed. The new allocation of resource along with expected number of fault removed, percentages of faults removed and faults remaining for each module after solving the problem (P2) through the problem (P5) is summarized in Table-2. The total number of faults that can be removed through this allocation is 146.8(i.e. 58.4% of the fault content is removed from the Software). In addition to the above if it is desired that a certain percentage of the total faults are to be removed then additional testing resources would be required. It is interesting to study this tradeoff and Table-3 summarizes results, where the required percentage of faults removed is 60%. To achieve this, 3000 units of additional testing effort is required. The total number of faults that can be removed through this allocation is 150.8(i.e. 60.09% of the fault content is removed from the Software). Analysis given in Tables-1, 2 and 3 help in providing the developer an insight into the resource allocation and the corresponding fault removal phenomenon and the objective can be set accordingly.

Table - 1

Module	a_i	b_i	X_i^*	m_i*	% of faults removed	% of faults remaining
1	63	5.33E-05	25435	46.7689	74.24	25.76
2	13	0.000252	5280.7	9.56979	73.61	26.39
3	6	0.000526	2459.5	4.3553	72.59	27.41
4	51	5.17E-05	21549	34.2571	67.17	32.83
5	15	0.000171	6354.5	9.93004	66.2	33.8
6	39	5.72E-05	16554	23.8778	61.23	38.77
7	21	9.94E-05	8857.2	12.2916	58.53	41.47
8	9	0.000174	3412.3	4.03476	44.83	55.17
9	23	5.06E-05	5845.6	5.88626	25.59	74.41
10	11	8.78E-05	1251.9	1.14528	10.41	89.59
Total	251		97000	152.117	60.6	39.4

Table-2

Module	a_i	a_{io}	Z_i*	Y_i*	$m_i(Y_i)$	m_i*	% of faults removed	% of faults remaining
1	63	32	13300	7495.5	10.2	42	67	33
2	13	7	3064.6	1235.6	1.61	8	66.21	33.79
3	6	3	1317.3	672.1	0.89	4	64.89	35.11
4	51	26	13793	2969.7	3.56	30	57.96	42.04
5	15	8	4464.8	440.4	0.51	8	56.71	43.29
6	39	20	12565	0	0	20	51.28	48.72
7	21	11	7465.7	0	0	11	52.38	47.62
8	9	5	4652.5	0	0	5	55.56	44.44
9	23	12	14586	0	0	12	52.17	47.83
10	11	6	8978.1	0	0	6	54.55	45.45
Total	251	130	84187	12813.3	16.8	146	58.48	41.52

Table-3

Module	a_i	a_{io}	Z_i*	Y_i*	$m_i(Yi)$	X_i*	m_i*	% of faults removed	% of faults remaining
1	63	32	13300	8624.7	11.74	21924.6	44	69.43	30.57
2	13	7	3064.6	1474.2	1.93	4538.82	9	68.7	31.3
3	6	3	1317.3	786.52	1.048	2103.79	4	67.47	32.53
4	51	26	13793	4134.5	5.13	17927.4	31	61.05	38.95
5	15	8	4464.8	793.16	0.984	5257.95	9	59.9	40.1
6	39	20	12565	0	0	12565.5	20	51.3	48.7
7	21	11	7465.7	0	0	7465.66	11	52.38	47.62
8	9	5	4652.5	0	0	4652.5	5	55.55	44.45
9	23	12	14586	0	0	14585.7	12	52.18	47.82
10	11	6	8978.1	0	0	8978.11	6	54.55	45.45
Total	251	130	84187	15813	20.8	100000	151	60.09	39.91

5. Conclusion

In this paper we have discussed a couple of optimization problems occurring during module testing phase of software development life cycle. A dynamic programming approach for finding the optimal solution has been proposed. Using simple recursion equations it is shown how fault removal for each module and that of the software can be maximized, by judicious allocation of resources. It is observed that after certain duration of testing, fault removal becomes difficult in the sense that greater effort will be required to remove each additional fault. As the reliability of software is of utmost importance scientific decision making is required while deciding the resource budget. The tradeoff as shown in section-4 can be useful in this regard.

Alternatively if the developer is not too keen on an optimal solution but is satisfied with an efficient solution, Goal Programming approach may be desirable in that case. We are further looking into this aspect.

Appendix:

Proof of the theorem-1

We have following recursion equations given in (7):

$$f_1(Z) = \max_{X_1 = Z} \left\{ a_1 (1 - e^{-b_1 X_1}) \right\}$$

$$f_n(Z) = \max_{0 \le X_n \le Z} \left\{ a_n (1 - e^{-b_n X_n}) + f_{n-1}(Z - X_n) \right\}, n = 2, \ldots, N$$

The above problem can be solved through forward recursion in N stages as follows.

Stage-1: Let $n=1$ then we have

$$f_1(Z) = \max_{X_1 = Z} \left\{ a_1 (1 - e^{-b_1 X_1}) \right\} = a_1 (1 - e^{-b_1 Z})$$

Stage-2: Let $n=2$ then we have

$$f_2(Z) = \max_{0 \le X2 \le Z} \left\{ a_2 (1 - e^{-b_2 X_2}) + f_1(Z - X_2) \right\}$$

Substituting $f_1(Z - X_2)$ in above we have

$$f_2(Z) = \max_{0 \le X2 \le Z} \left\{ a_2 (1 - e^{-b_2 X_2}) + a_1 (1 - e^{-b_1 (Z - X_2)}) \right\}$$

Now let, $F_2(X_2) = \left\{a_2(1-e^{-b_2X_2}) + a_1(1-e^{-b_1(Z-X_2)})\right\}$ then

$f_2(Z) = \max_{0 \le X2 \le Z} \{F_2(X_2)\}$

The maxima can be found through calculus.

$$\frac{dF_2(X_2)}{dX_2} = a_2b_2e^{-b_2X_2} - a_1b_1e^{-b_1(Z-X_2)}$$

The sufficiency condition can be checked through the second derivative condition:

$$\frac{d^2F_2(X_2)}{dX_2^2} = -a_2b_2^2e^{-b_2X_2} - a_1b_1^2e^{-b_1(Z-X_2)} \le 0$$

The following three situations can occur.

(i) $\frac{dF_2(X_2)}{dX_2} < 0$ (ii) $\frac{dF_2(X_2)}{dX_2} = 0$ (iii) $\frac{dF_2(X_2)}{dX_2} > 0$

If $\frac{dF_2(X_2)}{dX_2} < 0$, then X_2 =0.

At X_2 =0 $\frac{dF_2(X_2)}{dX_2} = a_2b_2 - a_1b_1e^{-b_1Z} < 0$

i.e. $1 \le e^{b_1Z} < \frac{a_1b_1}{a_2b_2}$

Which implies $a_1b_1 > a_2b_2$,in other words the detectability in module -1 is higher than module –2.

Similarly $\frac{dF_2(X_2)}{dX_2} > 0$ implies $X_2 = Z$

and we have $\frac{a_2b_2}{a_1b_1} > e^{b_2Z} \ge 1$

Hence $a_2b_2 > a_1b_1$, the testing resources would be allocated to module -2 first as the detectability is higher there.

Finally if $\frac{dF_2(X_2)}{dX_2} = 0$

$X_2^* = \frac{1}{b_1+b_2}\left\{b_1Z - \log\frac{a_1b_1}{a_2b_2}\right\}$, i.e. $X_2^* = \frac{1}{\mu_1+b_2}\left\{\mu_1Z - \log\frac{\mu_1V_1}{a_2b_2}\right\}$,

and $$f_2(Z) = \sum_{i=1}^{2} a_i - e^{-\frac{a_1 b_1}{a_2 b_2} Z} \left\{ a_2 \left(\frac{a_1 b_1}{a_2 b_2}\right)^{\frac{b_2}{b_1 + b_2}} + a_1 \left(\frac{a_2 b_2}{a_1 b_1}\right)^{\frac{b_1}{b_1 + b_2}} \right\}$$

i.e. $$f_2(Z) = \sum_{i=1}^{2} a_i - e^{-\mu_2 Z} V_2$$

Where $$\mu_1 = \frac{1}{\frac{1}{b_1}} = b_1,\ V_1 = a_1,\ \mu_2 = \frac{1}{\frac{1}{b_1} + \frac{1}{b_2}} = \frac{b_1 b_2}{b_1 + b_2}$$

$$V_2 = \left\{ a_2 \left(\frac{V_1 \mu_1}{a_2 b_2}\right)^{\frac{\mu_2}{\mu_1}} + V_1 \left(\frac{a_2 b_2}{V_1 \mu_1}\right)^{\frac{\mu_2}{b_2}} \right\}$$

Now proceeding by induction it can be shown for nth stage,

$$X_n^* = \frac{1}{\mu_{n-1} + b_n} \left\{ \mu_{n-1} Z - \log \frac{\mu_{n-1} V_{n-1}}{a_n b_n} \right\}$$

and $$f_n(Z) = \sum_{i=1}^{n} a_i - e^{-\mu_n Z} V_n \text{ for } n=1\ldots N$$

The proof is complete.

References

1. Goel A.L., Software Reliability Models: Assumptions, limitations and applicability, IEEE Trans. On software engineering, SE-11, pp. 1411-1423, 1985.
2. Hadley, G., Nonlinear and Dynamic Programming, Addision-Wesley, Reading Mass, 1964.
3. Ichimori, T, Yamada, S. And Nishiwaki M., Optimal allocation policies for testing-resource based on a Software Reliability Growth Model, Proceedings of the Australia –Japan workshop on stochastic models in engineering, technology and management, pp. 182-189, 1993.
4. Kapur P.K. and Garg R.B.; Cost reliability optimum release policies for a software system with testing effort, OPSEARCH, vol. 27, no. 2, pp. 109-118, 1990.
5. Kapur P.K., Garg R.B. and Kumar, S.; Contributions to Hardware and Software Reliability, World Scientific, Singapore, 1999.

6. Kapur, P.K. and Bardhan, A.K., Modelling, allocation and control of resources: an interdisciplinary approach in software reliability and marketing, Operations Research, Eds. M. Agarwal and K. Sen, Narosa Publishing House, New Delhi 2001.
7. Kubat P. and Koch H.S., Managing test procedures to achieve reliable software, IEEE Trans. On Reliability, Vol. R-32, pp. 299-303, 1983.
8. Musa J.D., Iannino A. And Okumoto K, Software Reliability-Measurement, Prediction and Application, Mc Graw Hill, 1987.
9. Yamada S. And Ohtera H. And Narihisa H., Software Reliability Growth Model with testing effort, IEEE Trans. On Reliability, vol. R-35, pp. 19-23, 1986.

KALMAN FILTERS AND ARMA MODELS

Aniello FEDULLO[1]

Abstract. The Kalman filter is the celebrated algorithm giving a recursive solution of the prediction problem for time series. After a quite general formulation of the prediction problem, the contributions of its solution by the great mathematicians Kolmogorov and Wiener are shorthly recalled and it is showed as Kalman filter furnishes the optimal predictor, in the sense of least squares, for processes which satisfy the linear models with a finite number of parameters, that are the ARMA models.

1. Introduction: Time Series

A *time series*, in our study, is considered like a finite part (*a sample*) of a single realization of a stochastic process. The fundamental problem of the anaysis of the time series is the following: given a time series, infer, al least in part, the characteristics of the process. Remember that, in general, a stochastic process is characterized by the joint distributions of all the finite sub-families of its random variables. If there is no other information known on the process, or if no other hypothesis is made about it, the problem is unsolvable or perhaps hill-posed. Conversely, if we limit ourselves to particular families of processes, the above-mentioned statistical inference is possible. In particular this happens for *weak (second order) stationary*, *ergodic*, *invertible* and with a *Gaussian residual* processes (see Papoulis 1965). In the following, unless otherwise indicated, we will consider only such processes.

2. The prediction problem

The information on the process inferred from the time series allows to resolve problems particularly important for applications, such as those of prediction, filtration and control. In the present work we will concentrate our

[1] Dipartimento di Matematica e Informatica, Università di Salerno, Italia.

attention on the first of the three. The problem of prediction is to estimate a future value *X(t+m) (m>0)* of a time series whose past values *X(t), X(t-1)* , …, *X(t-n)* are known. The estimation $\hat{X}^{(t+m)}$ of *X(t+m)* will be a suitable function of these latter values which minimizes

$$M(m):=E[(\hat{X}^{(t+m)}-X(t+m))^2].$$

It can be shown that in the absence of imposed constraints on the form of the aforementioned function, the preceding minimum problem has the solution given by the *conditional expectation*

$$E[X(t+m)\,|\,X(t),X(t-1),\ldots,X(t-n)].$$

The calculation of the latter, in general, requires the knowledge of the joint distribution of the random variables involved, that is a very detailed knowledge of the process, one rather difficult to arrive at. Observe that if the joint distributions are *normal* then the conditional expectation is linear, for which the so called *predictor* is written

$$\hat{X}^{(t+m)}=a_0X(t) + a_1X(t-1)+ \ldots + a_nX(t-n)$$

where a_0, …, a_n are constants (dependent only on *m* and *n*) to determine. This is done in such a way to minimize *M* and in order to do so it suffices to know the *autocovariance*. If the process is Gaussian then the linear predictor (2) is *optimal*, in the sense of least squares. Otherwise the conditional expectation is not in general linear and, as said, we could not know how to calculate it easily. Nevertheless, in such a case, we can restrict to the afore mentioned linear predictors and search among them for those which are optimal. The originators of the researches on linear predictors were Kolmogorov and Wiener in the early '40s. The publication of their works, however, was held up until 1949 because of military concerns (automatic poynting of anti-aircraft weapons and fire control). It became clear then that Kolmogorov and Wiener had resolved independently the same problem, using different techniques. They, as it was understood afterwards, related to different choises of the coordinates related up to the same geometric problem in Hilbert space. Furthemore, both the authors also studied the semi-infinite version of the problem ($n \to \infty$), for the major simplicity of the mathematical treatment. That, nevertheless, can be used as a good approximation of the real case where *n* is large but finite.

3. Recursive Algorithms

Kolmogorov's and Wiener's approaches to the problem of prediction has the advantage of furnishing an explicit expression of $X^{(t+m)}$, but a notable disadvantage is that it must be recalculated ex novo for different values, also consecutive, of *m*. This carries enormous complications in the computations, especially for real time applications. Such problems were overcome by the development of recursive algorithms. These calculate the estimation at time *n+1* by means of a simple correction of that at time n, with a notable saving of memory and of computation time. The two most famous recursive approaches are the one by Box and Jenkins and the one by Kalman. Both conducted to the optimal linear predictor in the sense of least squares, and both are applied only to the processes which satisfy the linear models with a finite number of parameters.

4. Linear Models with a Finite Numbers of Parameters

a) ARMA models

Let *X(t)* be a stochastic process which satisfies the properties indicated at the end of section 3 and which we assume, without loss of generality, has mean 0. Let *h(ω)* be the *spectral density* of *X(t)*; that is a non negative real function of ω which we will also assume to be a rational function of exp *(i ω)*. For the *spectral factorization theorem* there exist the polynomials *A* e *C* such that

$$h(\omega) = \text{cost}\ \frac{A(e^{i\omega})A^*(e^{i\omega})}{C(e^{i\omega})C^*(e^{i\omega})}$$

where

$$\mathrm{A}(z) := \sum_{i=0}^{p} a_i z^i, \qquad A^*(z) := \sum_{i=0}^{p} a_i z^{p-i}; \quad a_0 := 1; \quad a_i \in \mathbb{R} \ \text{ for every } \ i$$

$$C(z):=\sum_{i=0}^{q} c_i z^i, \quad C^*(z):=\sum_{i=0}^{q} c_i z^{q-i}; \quad c_0:=1; \quad c_i \in \mathbb{R} \text{ for every } i$$

We observe that *A(w) = 0* if and only if *A(1/w)=0* , by which, in virtue of the *fundamental theorem of algebra*, we can choose *A* and *C* in such a way that their zeros are all in modulus greater than 1.
Then the process *X(t)* can be written as

$$X(t)=\frac{A(\beta)}{C(\beta)} e(t)$$

where β is the *shift operator* defined as

$$\beta X(t):= X(t-1) \quad \text{for every } t$$

and *e(t)* is a *white noise*, that is a noise made up of random variables with mean *0*, variance σ^2 and pairwise uncorrelated. In fact it is easy to verify that the right hand of (4) has the spectrum (3). Multiplying both sides of (4) by *C* one has

$$C(\beta)X(t)=A(\beta)e(t)$$

called *autoregressive moving average* or ARMA model.

b) Models with State Space

They are described by equations of the type

$$s(t+1)=Fs(t)+Gu(t)+w(t), \qquad X(t)=Hs(t)+e(t)$$

where *s(t)* is a *n* dimensional vector called *state*, and *F, G, H* are matrices of adequate dimensions, *u(t)* is the *input* (which is considered relevant in the problems of control; here we may assume to be *0*), *w(t)* is the so-called *noise of process*, made up of random variables pairwise uncorrelated with covariance matrix $E[w(t)\tilde{w}(s)]=\delta_{ts} R_1$, *e(t)* is the so-called *noise of*

mausure, also made up of random variables pairwise uncorrelated, with covariance matrix $E[e(t)\tilde{e}(s)] = \delta_{ts} R_2$; between the two noises there is in general some correlation given by $E[w(t)\tilde{e}(s)] = \delta_{ts} R_{12}$. Beyond this, one assumes that $s(0)$, the *initial state*, is a random vector independent of the future terms of noise, with mean s and covariance matrix $\prod_0$. Observe that the model described above is general enough to include the multivariate case in which $X(t)$ has a dimension greater than 1. Even if it is possible to associate such a model to an ARMA model (eventually vectorial) and vice versa, the use of the state space is revealed more versatile and powerfull.

5. The Kalman Filter Algorithm

Let return to the recursive algorithms of the preceeding section 3. Box and Jenkins approach can be regarded as a special case of the most general and most powerfull algorithm of the Kalman filter(cfr. Caines 1972). Kalman's algorithm, based on the description of the linear model by means of the state space, lends itself to be extended to multivariate processes, with little additional strength, differently from Box and Jenkins approach. Consequently in the following we only illustrate Kalman's algorithm. We are considering for the model (6) the problem of estimating the state vector, given $X(t)$ (and possibly $u(t)$). Let

$$\hat{s}(t) := E[s(t) \mid X(0), u(0), \ldots, X(t-1), u(t-1)].$$

In the case the initial state and the involved noises are Gaussian, the preceding estimation is obtained by the following recursive procedure (*Kalman filter*):

$$\hat{s}(0) := s(0)$$
$$\hat{s}(t+1) = F\hat{s}(t) + Gu(t) + K(t)[X(t) - H\hat{s}(t)].$$

The matrix $K(t)$ is called *Kalman gain* and is given by

$$K(t) := [FP(t)\tilde{H} + \tilde{R}_{12}][HP(t)\tilde{H} + R_2]^{-1}$$

where the matrix $P(t)$ is a solution of the *Riccati equation*

$$P(0) := \Pi_0$$

$$P(t+1)=FP(t)\tilde{F}+R_1-[FP(t)\tilde{H}+\tilde{R}_{12}][HP(t)\tilde{H}+R_2]^{-1}[FP(t)\tilde{H}+\tilde{R}_{12}].$$

In general the algorithm, when the disturbances are not Gaussian, does not furnish an estimation coinciding with the conditional expectation (7); but a minimum covariance estimation among those which are linear in X and u. Finally we can note that Kalman filter works also in the most general cases where all matrices are time-dependent; but a detailed study of this would be beyond the imposed limits of the present paper.

Bibliography

- Box G.E.P. e G.M. Jenkins (1970), *Time series Analysis, Forecasting and Control*, Holden-Day, San Francisco.
- Caines P.E. (1972), *Relationship between Box-Kenkins-Astrom control and Kalman linear regulator*, Proc. IEEE, **119**, 615--620.
- Kalman R.E. (1960), *A new approach to linear filtering and prediction problems*, Trans. ASME J. Basic Engrg., ser. D, **82**, 35--45.
- Kalman R.E. e R.S. Bucy (1961), *New results in linear filtering and prediction problems*, Trans. ASME J. Basic Engrg., ser. D, **83**, 95--108.
- Kolmogoroff A.N. (1941), *Interpolation und extrapolation von stationaren Zufallingen Folgen*, Bull. Acad. Sci. (Nauk), U.S.S.R., ser. math., **5**, 3--14.
- Papoulis A. (1965), *Probability, random variables and stochastic processes*, Mc Graw Hill, New York.
- Wiener N. (1949), *The extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications*, Wiley, New York.

MODERATE-DENSITY BURST ERROR CORRECTING LINEAR CODES

Bal Kishan Dass and Gangmei Sobha*
Department of Mathematics, University of Delhi, Delhi-110007, India

ABSTRACT

Lower and upper bounds for the existence of linear codes which correct burst of length b (fixed) and whose weight lies between certain limits have been presented.

Keywords : Error detecting codes, error correcting codes, burst errors, moderate-density burst, lower and upper bounds.

1. INTRODUCTION

It is well known that during the process of transmission errors occur predominantly in the form of a burst. However, it does not generally happen that all the digits inside any burst length get corrupted. Also when burst length is large then the actual number of errors inside the burst length is also not very less. Keeping this in view, we study codes which detect/correct moderate-density burst errors.

In the literature, various kinds of burst errors have been studied, viz. open loop bursts (c.f. Peterson and Weldon, Jr. (1972), p.109), closed-loop bursts [Campopiono, 1962], C.T. bursts [Chien and Tang, (1965)], low-density bursts [Dass, 1975], etc. One important kind of bursts errors which has not drawn much attention is burst of specified length (fixed) [Dass, 1982]. In this paper, we derive lower and upper bounds for linear codes that detect/correct Moderate-density bursts of length b (fixed) for some positive integer b.

In what follows we shall consider a linear code to be a subspace of n-tuples over GF(q). The weight of a vector shall be considered in the Hamming's sense [Hamming, 1950] and we shall mean by a burst of length b (fixed), is an n-tuple whose only nonzero components are confined to b consecutive positions, the first of which is nonzero and the number of its starting positions is the first (n-b+1) components.

* On Study leave from Department of Mathematics and Statistics, College of Agriculture, Central Agricultural University, Imphal-795001, India.

2. BOUNDS FOR CODES CORRECTING MODERATE-DENSITY BURSTS

In this section, we consider codes correcting moderate-density burst errors. We first obtain a lower bound on the number of check digits which is a necessary conditions for the existence of codes capable of correcting bursts of length b (fixed) with weight lying between w_1 and w_2 ($0 \le w_1 \le w_2 \le b$).

Before this we prove the following Lemma.

Lemma 1. If J (n, b, w_1, w_2) denotes the number of n-tuples over GF(q) which form bursts of length b (fixed) with weight lying w_1 and w_2 ($0 \le w_1 \le w_2 \le b$) then

$$J(n, b, w_1, w_2) = (n - b + 1) \sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i+1} \qquad (1)$$

Proof. The Lemma follows immediately from the fact that the number of bursts of length b (fixed) with weight i is

$$\binom{b-1}{i-1}(q-1)^i (n-b+1). \; \square$$

Theorem 1. The number of parity check symbols in an (n, k) linear code that corrects all bursts of length b (fixed) of weight lying between w_1 and w_2 ($0 \le w_1 \le w_2 \le b$) is at least.

$$\log_q [1 + J (n, b, w_1, w_2)]. \qquad (2)$$

Proof. Since the code has q^{n-k} cosets in all, and all the error patterns are to be in different cosets of the standard array, therefore, in view of Lemma 1, we must have

$$q^{n-k} \ge 1 + J (n, b, w_1, w_2). \qquad (3)$$

The result now follows by taking logarithm on both sides. □

Remarks. If we put $w_1 = 0$ and $w_2 = b$ in the above result then weight constraints imposed on the burst becomes redundant and we get

$$q^{n-k} \ge 1 + [(n-b+1] (q-1)] q^{b-1},$$

which gives the number of parity check digits in an (n, k) linear code over GF(q) that corrects all bursts of length b (fixed), a result due to Dass [1980].

Now, if we take, $w_1 = 0$ and $w_2 = w$ in (3) we get

$$q^{n-k} \ge 1 + (n-b+1) (q-1) [1+(q-1)]^{(b-1, w-1)},$$

which gives the number of check digits required for linear codes correcting bursts of length b (fixed) with weight w or less ($w \le b$) a result which is again due to Dass [1983].

Moreover, when $w_1 = w$ and $w_2 = b$ we obtain

$$q^{n-k} \ge 1 + (n-b+1) \sum_{\substack{i=w-1 \\ w \ne 0}}^{b-1} \binom{b-1}{i} (q-1)^{i+1},$$

which gives the number of check digits required in an (n, k) linear code that corrects all bursts of length b (fixed) with weight w or more ($w \le b$) which coincides with the result due to the authors [2000].

Now, we first obtain a sufficient condition giving an upper bound for the existence of a code capable of detecting moderate-density burst errors, and then in the theorem following this result we shall obtain an upper bound for codes correcting such errors.

Theorem 2. Given non-negative integers, w_1, w_2 and b such that $0 \le w_1 \le w_2 \le b$, a sufficient condition that there exists an (n, k) linear code that has no burst of length b (fixed) whose weight lies between w_1 and w_2, as a code word is

$$q^{n-k} > 1 + \sum_{\substack{i=w_1-1 \\ w_1 \ne 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i} \quad . \tag{4}$$

Proof. The existence of such a code will be proved by constructing a suitable (n–k) x n parity check matrix H for the desired code. For this we first construct an (n–k) x n matrix H′ and then H will be obtained by reversing altogether the columns of H′.

We select any non-zero (n–k)-tuple as the first column of H′. Subsequent columns are added to H′ in such a way that after having selected j–1 columns $h_1, h_2, \ldots, h_{j-1}$ suitably a nonzero (n–k)-tuple is chosen as the j-th column such that it is not a linear combination of any p columns ($w_1 - 1 \le p \le w_2 - 1$) from the immediately preceding b–1 columns $h_{j-b+1}, h_{j-b+2}, \ldots, h_{j-1}$. Such a condition will ensure that a burst of length b (fixed) with weight lying between w_1, w_2 cannot be a code word in the code whose parity-check matrix is H to be obtained from H′ as prescribed earlier. In other words,

$$h_j \ne a_1 h_{j-b+1} + a_2 h_{j-b+2} + \ldots + a_{b-1} h_{j-1}, \tag{5}$$

where number of nonzero a_i's lies between $w_1 - 1$ and $w_2 - 2$. Since $a_i \in GF(q)$, the possible number of linear combinations on the R.H.S. of (5) including the case when all the α_i's are zero is

$$1 + \sum_{\substack{i=w_1-1 \\ w_1 \ne 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i} \quad .$$

Therefore, a column h_j can be added to H′ provided that this number is less than the total number of (n–k)-tuples.

At worst, all these linear combinations might yield a distinct sum, therefore, h_j can always be added to H′ provided that

$$q^{n-k} > 1 + \sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^i \quad . \tag{6}$$

It is important note that this relation is independent of j, therefore we can go on adding the columns as long as we wish but for the code of length j we shall stop after choosing j columns. So for j = n we shall added upto n columns.

By reversing the order of columns of the matrix $H' = [h_1, h_2, \ldots, h_n]$, we get the required parity check matrix $H = [H_1, H_2, \ldots, H_n]$ where $h_I = H_{n-i+1}$ (i.e. $h_n = H_1$, $H_{n-1} = H_2$, …, $h_1 = H_n$].

Thus we obtain the inequality as stated in (4). □

Examples 1. Consider the following 5x7 matrix of a (7, 2) code over GF (2).

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

This matrix has been constructed by the synthesis procedure outlined in the proof of theorem 2 by taking b = 4, $w_1 = 2$ and $w_2 = 3$. The code words of this code are 0000000, 0111101, 1000111, 1111010 which are not bursts of length 4 with weight lying between $w_1 = 2$, $w_2 = 3$.

Next we derive sufficient condition for codes correcting moderate-density bursts of length b (fixed).

Theorem 3. A sufficient condition for the existence of an (n, k) linear code over GF(q) which corrects all burst of length b (fixed) with weight lying between w_1 and w_2 ($0 \leq w_1 \leq w_2 \leq b$) is

$$q^{n-k} > 1 + \left[\sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^i \right] \left[(n-2b+1) \sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i+1} \right] + \left[\sum_{i=0}^{p} \binom{b-1}{i} (q-1)^i \right]$$
$$+ \sum_{k=1}^{b-1} \left[\sum_{\substack{r_1=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-k}{r_1} (q-1)^{r_1+1} \sum_{\substack{r_2,r_3: \\ r_2+r_3 \geq w_1-2}} \binom{k-1}{r_2} \binom{b-k-1}{r_3} (q-1)^{r_2+r_3} \right]$$
$$+ \sum_{k=1}^{b-1} \left[\sum_{\substack{r_2,r_2,r_3: \\ r_1+r_2+r_3 \leq 2w_2-2 \\ w_1-k-1 \leq r_1 \leq w_1-2}} \binom{b-k}{r_1} \binom{k-1}{r_2} \binom{b-k-1}{r_3} (q-1)^{r_1+r_2+r_3+1} \right] \tag{7}$$

where $p = 2w_2 - 1$, when $b \geq 2w_1$, $q=2$

$= 2b - 2w_1 - 1$ when $b < 2w_1$, $q = 2$

and $w_1 - k \leq r_1 \leq w_2 - 1$, $w_1 - k - 1 \leq w_2 - 1$, $0 \leq r_2 \leq 2w_2 - 3$, $r_1 + r_2 + r_3 \leq 2w_2 - 2$.

Proof. The existence of such a code shall be proved as in the previous theorem.

A nonzero (n-k)-tuple is chosen as the first column of H′. Subsequent columns are added such that after having selected j–1 columns, $h_1, h_2, \ldots, h_{j-1}$ suitably a column h_j is added provided that it is not a linear combination of any number of columns lying between w_1-1 and w_2-1 among the immediately preceding b–1 columns $h_{j-b+1}, h_{j-b+2}, \ldots, h_{j-1}$ together with any number of columns lying between w_1 and w_2 among any b consecutive columns out of all the j–1 columns selected so far. In other words, h_j can be added provided that

$$h_j \neq (\alpha_1 h_{j-b+1} + \alpha_2 h_{j-b+2} + \ldots + \alpha_{b-1} h_{j-1}) + (\beta_i h_i + \beta_{i+1} h_{i+1} + \ldots + \beta_{i+b} h_{i+b-1}) \tag{8}$$

where h_j's are any b consecutive columns from all the j–1 previously chosen columns and the number of nonzero β_i's lies between w_1 and w_2 whereas the number of nonzero α_i's lies between w_1-1 and w_2-1 along with the case when all the α_i's are zero.

To compute the number of all possible linear combinations corresponding to R.H.S. of (8) for all possible choices of α_j and β_i we analyse three different cases as follows.

Case 1. When h_j's are completely confined to the first j–b columns.

The number of ways that the coefficients α_j's can be selected is

$$\sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^i \ . \tag{9}$$

Further the number of ways that the coefficients of β_i's which form a burst of length b (fixed) with weight lying between w_1 and w_2 in a vector of length j-b can be selected is (refer Lemma 1).

$$J(j-b, b, w_1, w_2) = (j-2b+1) \sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i+1} \quad . \tag{10}$$

Therefore, the total number of choices of coefficients in this case is

$$\left[\sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i} \right] \left[(j-2b+1) \sum_{\substack{i=w_1-1 \\ w_1 \neq 0}}^{w_2-1} \binom{b-1}{i} (q-1)^{i+1} \right] . \tag{11}$$

Case II. When h_j's are completely confined to the immediately preceding b–1 columns.

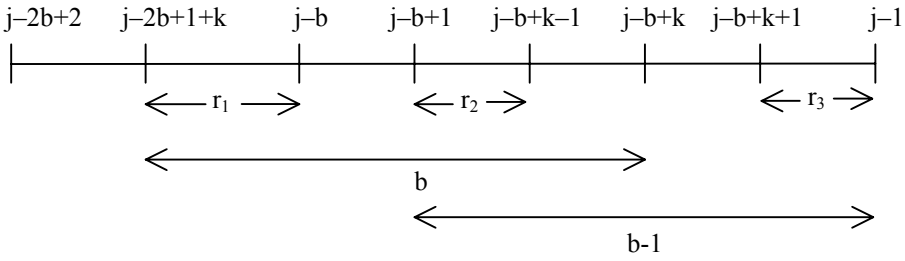
In this case the number of linear combinations corresponding to R.H.S. of (8) is

$$\sum_{i=0}^{p} \binom{b-1}{i} (q-1)^{i} \quad , \tag{12}$$

where $p = 2w_2-1$, when $b \geq 2w_1$, $q=2$

$= 2b-2w_1-1$ when $b < 2w_1$, $q=2$

Case III. When h_j's are neither completely confined to the first (j–b) columns nor to the last b–1 columns.



Let the burst starts from (j–2b+1+k)-th position which can continue upto (j–b+k)-th position, ($1 \leq k \leq b-1$). We select at least w_1-1 and at the most w_2-1 nonzero components corresponding to j–2b+1+k, j–2b+2+k, …, j–b+k–1 columns together with nonzero components lying between w_1-1 and w_2-1 corresponding to j–b+1, j–b+2, …, j–1 columns. Let r_1, r_2 and r_3 be the number of nonzero components corresponding to columns lying between (j–2b+1+k)-th to (j–b)-th, (j–b+1)-th to (j–b+k–1)-th and (j–b+k+1)-th to (j–1)-th column respectively, such that

$w_1 - k \le r_1 \le w_2 - 1,\ w_1 - k - 1 \le r_3 \le w_2 - 1,\ 0 \le r_2 \le 2w_2 - 3,\ r_1 + r_2 + r_3 \le 2w_2 - 2 \qquad (13)$

Therefore total possible number of distinct choices of coefficients is

$$+\sum_{k=1}^{b-1}\left[\sum_{\substack{r_1=w_1-1\\ w_1\neq 0}}^{w_2-1}\binom{b-k}{r_1}(q-1)^{r_1+1}\sum_{\substack{r_2,r_3:\\ r_2+r_3\ge w_1-2}}^{w_2-1}\binom{k-1}{r_2}\binom{b-k-1}{r_3}(q-1)^{r_2+r_3}\right]$$

$$+\sum_{k=1}^{b-1}\left[\sum_{\substack{r_2,r_2,r_3:\\ r_1+r_2+r_3\le 2w_2-2\\ w_1-k-1\le r_1\le w_1-2}}\binom{b-k}{r_1}\binom{k-1}{r_2}\binom{b-k-1}{r_3}(q-1)^{r_1+r_2+r_3+1}\right] \qquad (14)$$

Thus total possible number of distinct linear combinations corresponding to (8), which cannot be equal to h_j including zero vector is

$$1+\left[\sum_{\substack{i=w_1-1\\ w_1\neq 0}}^{w_2-1}\binom{b-1}{i}(q-1)^i\right]\left[(n-2b+1)\sum_{\substack{i=w_1-1\\ w_1\neq 0}}^{w_2-1}\binom{b-1}{i}(q-1)^{i+1}\right]+\left[\sum_{i=0}^{p}\binom{b-1}{i}(q-1)^i\right]$$

$$+\sum_{k=1}^{b-1}\left[\sum_{\substack{r_1=w_1-1\\ w_1\neq 0}}^{w_2-1}\binom{b-k}{r_1}(q-1)^{r_1+1}\sum_{\substack{r_2,r_3:\\ r_2+r_3\ge w_1-2}}\binom{k-1}{r_2}\binom{b-k-1}{r_3}(q-1)^{r_2+r_3}\right]$$

$$+\sum_{k=1}^{b-1}\left[\sum_{\substack{r_1,r_2,r_3:\\ r_1+r_2+r_3\le 2w_2-2\\ w_1-k-1\le r_1\le w_1-2}}\binom{b-k}{r_1}\binom{k-1}{r_2}\binom{b-k-1}{r_3}(q-1)^{r_1+r_2+r_3+1}\right] \qquad (15)$$

Therefore, the j-th column can be added to H′ provided that

$$q^{n-k} > M, \qquad (16)$$

where M denotes expression (15).

For the existence of an (n, k) desired code relation (16) should hold for j = n so that it is possible to add upto nth column to form an (n–k) x n matrix. Thus we have constructed the matrix $H' = [h_i]$, (h_i denotes the i-th column from which we obtain the required parity check matrix $H = [H_i]$, (H_i denotes the i-th column) by reversing its column altogether, i.e. $h_i \rightarrow H_{n-i+1}$. This proves the result. □

Remarks 1. If we take $w_1 = 0$, $w_2 = b$ in (16) the weight constraints becomes redundant. Hence the bound gives $q^{n-k} > q^{b-1}\,[q^{b-1}\,(n–2b+1)\,(q–1)+1]$ which is a result due to Dass [1980].

2. If we put $w_1 = 0$, $w_2 = w$, in (16) we get the bound obtained by Dass [1982], which is a sufficient condition for the existence of low-density burst correcting code that corrects all bursts of length b (fixed).

Example 2. Consider the following matrix 6x9 of a (9,3) code over GF(2) which can correct all bursts of length 4 (fixed) with weight 2 or 3.

$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

This matrix is constructed by the synthesis procedure outline in the proof of theorem 3.

It can be seen from the table 1 that syndromes of all the correctable error patterns are distinct and therefore the null space of this matrix gives the desired code.

Table 1

Error Pattern	Syndrome	Error Pattern	Syndrome
110000000	000011	111000000	000111
011000000	000110	011100000	001110
001100000	001100	001110000	011100
000110000	011000	000111000	111000
000011000	110000	000011100	001111
000001100	011111	000001110	011110
101000000	000101	110100000	001101
010100000	001010	011010000	011010
001010000	010100	001101000	110100
000101000	101000	000110100	100111
000010100	101111	000011010	110001
000001010	100001	000001101	010101
100100000	001001	101100000	001011
010010000	010010	010110000	010110
001001000	100100	001011000	101100
000100100	110111	000101100	010111
000010010	010001	000010110	101110
000001001	101010	000001011	101011

REFERENCES

1. CHIEN, R.T. and D.T. TANG (1965) : On definition of a Burst, IBM J. Research & Develop., 9(4), 292-293.

2. DASS, B.K. (1975) : A sufficient Bound for Codes Correcting Bursts with Weight Constraints, Journal of the Association for Computing Machinery (J. ACM), Vol.22, No.4, 501-503.

3. DASS, B.K. (1980) : On a Burst-Error Correcting Linear Codes, J. Infor. & Opt. Sciences, Vol.1, No.3, 291-295.

4. DASS, B.K. (1983) : Low Density Burst-Error Correcting Linear Codes, Adv. in Mngt. Studies, Vol.2, pp.375-385.

5. DASS, B.K. and G. SOBHA (2000) : High-Density Burst Error Correcting Linear Codes, unpublished.

6. HAMMING, R.W. (1950) : Error Detecting and Error Correcting Codes, Bell System Tec. J., 29, 147-160.

7. PETERSON, W.W. and E.J. WELDON, Jr. (1972) : Error-Correcting Codes, Cambridge, Mass : The M.I.T. Press.

8. SHARMA, B.D. and S.N. GUPTA (1975) : On the existence of Moderate-Density Burst Codes, Journal of Mathematical Sciences, Vol.10, 8-12.

Cooperative Games, Finite Geometries and Hyperstructures

Antonio Maturo
Università di Chieti - Pescara

Abstract In this paper some relations between finite geometric spaces and cooperative games are considered. In particular by some recent results on blocking sets we have new results on blocking coalitions. Finally we introduce a new research field on the possible relations between quasihypergroups and cooperative games.

Keywords Cooperative Games, Finite Geometries, Blocking sets, Quasihypergoups

1. Cooperative games

Let M = {1, 2, …, n} be a finite non-empty set, called the *set of players*.
A function v: $\wp(M) \to R$ such that:

(C1) $v(\varnothing) = 0$;
(C2) (*superadditivity*) $\forall A, B \in \wp(M), (A \cap B = \varnothing) \Rightarrow v(A \cup B) \geq v(A) + v(B)$;

is called *characteristic function* on M.
The pair (M, v) is called *cooperative game with n players* and the subsets of M are called *coalitions.*

For every $A \in \wp(M)$ the number v(A) is the total gain that the players of A can have certainly forming a coalition, independently on the actions of the players not belonging to A. We assume the condition of "*side payment*", that is in every coalition A any player can transfer an amount of his gain to another player belonging to A and so it is important only the total gain of the coalition.
The condition (C2) is a consequence of the fact that the total gain obtained with an alliance between two disjoint coalitions is not inferior to the one without cooperation.

We write v(i) to denote v({i}). By (C2) it follows that in a cooperative game (M, v) we have $v(M) \geq \Sigma_{i \in M} v(i)$. If $v(M) > \Sigma_{i \in M} v(i)$ the game (M, v) is said to be *essential*, if the equality holds (M, v) is *inessential*.
It is easy to prove that a cooperative game is inessential if and only if:

(AD) (*additivity*) $\forall A, B \in \wp(M), (A \cap B = \varnothing) \Rightarrow (v(A \cup B) = v(A) + v(B))$

and so there are no advantages by the cooperation.

Two cooperative games (M, v) and (M, v'), with the same set of n players, are called *strategically equivalent*, we write (M, v) ≈ (M, v') if there exist n+1 real numbers k>0 and $c_1, c_2, \ldots, c_n$ such that:

(SE) $\forall A \in \wp(M),\ v'(A) = k\, v(A) + \Sigma_{i \in A}\, c_i$.

We obtain the game (M, v') by the game (M, v) with an initial payment c_r to any player r and by multiplying the total gain of any coalition by the scale factor k. Then we can assume the same strategies to solve (M, v) or (M, v').

Proposition 1.1. Let (M, v) a cooperative game. The system with n+1 equations and n+1 unknowns k > 0 and $c_1, c_2, \ldots, c_n$:

(ESI) $k\, v(M) + \Sigma_{i \in M}\, c_i = 1, \qquad k\, v(i) + c_i = 0,\ i \in M$

has determinant $v(M) - \Sigma_{i \in M}\, v(i)$ and so has solutions if and only if (M, v) is essential. In this case $k = 1/(v(M) - \Sigma_{i \in M}\, v(i))$ and so k > 0.
The system:

(NES) $k\, v(M) + \Sigma_{i \in M}\, c_i = 0, \qquad k\, v(i) + c_i = 0,\ i \in M$

has not trivial solution if and only if $v(M) - \Sigma_{i \in M}\, v(i) = 0$.

The relation ≈ is an equivalence relation among the cooperative games with the same set of players M. By proposition 1.1 we have that, for any equivalence class K with respect to ≈, we have a unique cooperative game (M, v)∈K, called *normal element* of K or *normal form* of the elements of K, such that v(i) = 0, ∀i∈M and v(M)∈{0, 1}. Precisely, v(M) = 1 if the game is essential and v(M) = 0 if it is inessential. The inessential games are in the same equivalence class and the normal form is such that v(A) = 0, $\forall A \in \wp(M)$. On the contrary, for n > 2, the essential games are in different classes.

2. Simple cooperative games and projective spaces

Let (M, v) be an essential cooperative game in normal form. Then v(i) = 0, ∀i∈M, and v(M) = 1. We say that (M, v) is a *simple game* if, $\forall A \in \wp(M)$, v(A)∈{0, 1}.
By (C2), for any coalition A, if A^c = M-A, we have three possibility:

(a) $v(A) = 1$ and $v(A^c) = 0$;
(b) $v(A) = 0$ and $v(A^c) = 1$;
(c) $v(A) = 0$ and $v(A^c) = 0$.

The set A is called *winning coalition* if (a) holds and *losing coalition* if (b) holds. It is evident that M is a winning coalition and the complement of a winning coalition is a losing coalition. So the number of winning coalitions is equal to the number of losing coalitions. The set A is said to be a *blocking coalition* if (c) holds. If A is a blocking coalition then also A^c is a blocking coalition. So, if there exist blocking coalitions, their number is even.
We have the following:

Proposition 2.1 Let W be a subset of a set $\wp(M)$, with M set of players. Then W is the set of the winning coalitions of a simple cooperative game (M, v) if and only if satisfy the following properties, called the "*axioms of Shapley*" (see [32], [34]):

(W1) $M \in W$;
(W2) $\forall A, B \in \wp(M), (A \in W, A \subseteq B) \Rightarrow B \in W$;
(W3) $\forall A \in \wp(M), A \in W \Rightarrow A^c \notin W$.

Proof. Let (M, v) be a simple cooperative game in normal form and let W be the set of winning coalitions. Then (W1) and (W2) are trivial and (W3) follows by (C2).
On the converse, let W be a subset of $\wp(M)$ satisfying the axioms of Shapley. We put, for any $A \in \wp(M)$, v(A) = 1 if $A \in W$ and v(A) = 0 otherwise. The pair (M, v) is a simple cooperative game and W is the set of winning coalitions.

By previous proposition, in the sequel we consider a simple cooperative game indifferently as the pair (M, v) or the pair (M, W).
All the properties of the losing coalitions are obtained from the ones of the winning coalitions by replacing any coalition A with A^c and $\subseteq$ with $\supseteq$ and vice versa.
We can prove the following:

Proposition 2.2 Let (M, W) be a simple cooperative game. A family Θ of subsets of M is the set of blocking coalitions of (M, W) if and only if:

(BC) $\forall X \in \Theta, \forall A \in W, X \cap A \neq \emptyset, X^c \cap A \neq \emptyset$.

Now we introduce some useful definitions.

Definition 2.1 Let M be a non empty set and let $\Im$ be a family of subsets of M.
We say that $\Im$ has the "*intersection property*" if we have:

(IP) $\forall A, B \in \Im, A \cap B \neq \emptyset$.

We say that $\Im$ has the "*non-inclusion property*" if we have:

(NI) $\forall A, B \in \Im, A \cap B^c \neq \emptyset$ and $A^c \cap B \neq \emptyset$.

Definition 2.2 Let M be a non empty set and let Φ and $\mathfrak{I}$ be two families of subsets of M. We say that "$\mathfrak{I}$ *is a generator of* Φ" or "Φ *is the closure* of $\mathfrak{I}$", and we write $\Phi = K(\mathfrak{I})$, if

(GK) $\Phi = \{A \in \wp(M): \exists B \in \mathfrak{I} / B \subseteq A\}$.

We say that "$\mathfrak{I}$ *is a minimal generator of* Φ" if $\mathfrak{I}$ has the *non-inclusion property* and is a *generator* of Φ.

By (W3) it follows that, in a simple cooperative game (M, W), W satisfies the intersection property. The following proposition shows that any family $\mathfrak{I}$ of subsets of M that has the intersection property generates the winning coalitions of a simple cooperative game.

Proposition 2. 3 Let M be a n-set, whose elements are called *players*, and let $\mathfrak{I}$ be a family of subsets non-void of M satisfying the intersection property. Then if W is the closure of $\mathfrak{I}$, the pair (M, W) is a simple cooperative game with W set of winning coalitions, called "*the game generated by* (M, $\mathfrak{I}$)".

Proof. Let W be the closure of $\mathfrak{I}$. Then (W1) and (W2) are evident. If $A \in W$, then there exists $B \in \mathfrak{I}$ such that $A \supseteq B$ and so $A^c \cap B = \varnothing$. Then, $\forall C \in \mathfrak{I}$, A^c don't contains C. Otherwise A^c must contain $C \cap B \neq \varnothing$, a contradiction. It follows that $A^c \notin W$.

Now we examine some relations between the cooperative simple games and the geometric spaces.

Definition 2. 3 A geometric space is a pair (M, Δ), with M a non-empty set, called *the support* and Δ a non-empty family of subsets of M. The elements of M are called *points* and the ones of Δ are called *blocks*. If any block has *at least two points* and any two blocks have *at most one point in common* (M, Δ) is called "*space of lines*" and the blocks are called also *lines*.
(M, Δ) is *non-degenerate* if there are at least two blocks.

Definition 2.4 A *projective space* is a geometric space (M, Δ) such that (see [7]):

(PS1) $\forall P, Q \in M$, $P \neq Q$, there is exactly one block containing {P, Q}, called *the line* PQ;
(PS2) (*Veblen-Young axiom*) Let A, B, C, D four distinct points such that AB intersects CD. Then AC intersects BD.
(PS3) Any line contains at least three points.

A non-degenerate projective space is a *projective plane* (or projective space with dimension 2) if the axiom (PS2) is replaced by the stronger axiom:

(PS2S) Two lines have at least a point in common.

If a non-degenerate projective space (M, Δ) is not a projective plane, for any A, B, C$\in$M, distinct and such that C not belongs to the line AB, we define "*plane* ABC", or "2-*dimensional subspace* ABC" of M, the union of the lines CX, with X$\in$AB.
We say that (M, Δ) has dimension 3 if:

(PSD3) A line and a plane have at least a point in common.

For recurrence we can consider projective spaces and subspaces with greater dimensions.

If (M, Δ) is a projective plane we have that Δ satisfies both the *intersection property* and the *non-inclusion property***.** So, by proposition 2.3, we have the following:

Proposition 2.4 Let (M, Δ) be a finite projective plane and let W be the closure of Δ. Then (M, W) is a simple cooperative game, with W set of winner coalitions, and Δ is a minimal generator of W.

If (M, Δ) is a projective space of dimension 3 or 4 the planes have both the *intersection property* and the *non-inclusion property*. Then we have:

Proposition 2. 5 Let (M, Δ) be a finite projective n-dimensional space with n$\in${3, 4} and let Δ* be the set of all the planes of M.
If W is the closure of Δ* then (M, W) is a simple cooperative game, with W set of winner coalitions, and Δ* is a minimal generator of W.

Definition 2.5 Let (M, Δ) be a geometric space and let $\Im$ be a family of subsets of M. A subset X of M is called a *blocking set* with respect to $\Im$ if.

(BS) $\forall A\in\Im$, $X\cap A \neq \varnothing$ and $X^c\cap A \neq \varnothing$.

If C is a subset of M containing a A$\in\Im$, by (BS) we have: $X\cap C \neq \varnothing$ and $X^c\cap C \neq \varnothing$. Then it follows the:

Proposition 2. 6 Let (M, Δ) be a geometric space, $\Im$ be a family of subsets of M and Φ be the closure of $\Im$. Then X is a blocking set with respect to $\Im$ if and only if it is a blocking set with respect to Φ.

Some corollaries of the previous propositions are:

Proposition 2. 7 Let (M, Δ) be a geometric space such that Δ has the intersection property. Then the blocking sets with respect to Δ are the blocking coalitions of the simple cooperative game (M, W), with W closure of Δ.

Proposition 2. 8 In a finite projective plane (M, Δ) the blocking sets with respect the lines are the blocking coalitions of the simple cooperative game (M, W), with W closure of Δ.

Proposition 2. 9 In a finite 3-dimensional or 4-dimensional projective space (M, Δ) the blocking sets with respect the planes are the blocking coalitions of the simple cooperative game (M, W), with W closure of Δ*, set of the planes.

The previous propositions show the importance of the research of blocking sets in a finite projective space.

In particular we have the fundamental problems to find:

(a) the *minimal* or *maximal* blocking sets;
(b) the *spectrum* of the minimal blocking sets, that is the set of all the possible cardinalities of the minimal blocking coalitions;
(c) the *minimal winning coalitions*;
(d) the *winning coalitions containing blocking coalitions.*

By (BS) it follows that the complement of a blocking set is also a blocking set, so to find the maximal blocking sets is equivalent to find the minimal ones.
Now we show some results in the particular case of projective planes.

It is well known that, in a non-degenerate finite projective plane, all the lines have the same number of points. If q+1 is such number, the projective plane is said to be of *order* q and is noted π_q. Moreover, the lines through a fixed point P are also q+1 and the points of π_q are $q^2 + q + 1$.
By (PS3) we have $q \geq 2$. It is well known (see [7], [17]) that there exists a Desarguesian projective plane if and only if q is a prime or a power of a prime and such plane is unique. The first value of q with non-Desarguesian planes is q = 9.

For small values of q we have:

- in π_2 there are not blocking sets;
- in π_3 there are exactly two blocking sets;
- the blocking sets on π_4 and π_5 are classified, respectively, in papers of Berardi - Eugeni ([2]) and Berardi - Innamorati ([5]);
- the blocking sets on π_7 are classified in papers of Innamorati and Maturo (see [23], [24], [25]). If k is the cardinality of a minimal blocking set on π_7 we have $12 \leq k \leq 19$. In particular there are, up to isomorphism, only two

minimal blocking sets of order 12 and there is only a minimal blocking set with 19 points.

In the general case there are the following results (Innamorati – Maturo, [23], [25]):

Proposition 2.10 Let S(q) the spectrum of the minimal blocking sets in π_q. Then, if $q \geq 4$, $S(q) \supseteq [2q-1, 3q-5] \cup \{3q-3\}$ and, if π_q is Desarguesian, $S(q) \supseteq [2q-1, 3q-3]$.

Proposition 2.11 A sufficient condition for the existence of a minimal blocking set with 3q–4 points on a non-Desarguesian plane π_q is that π_q contains a proper subplane of order two.

In [29] H. Newmann conjectured that any finite non-Desarguesian plane contains a proper subplane of order two. By previous proposition, if the conjecture is true, we have that also for the non-Desarguesian plane of order q there exists a blocking set with 3q – 4 points.

3. Cooperative games and finite geometric spaces

We introduce the following:

Definition 3.1 Let M be a non-void set and let Ψ and $\Im$ be two families of subsets of M. We say that "$\Im$ *is a intersection-generator of* Ψ" or "Ψ *is the intersection-closure* of $\Im$", and we write $\Psi = IK(\Im)$, if

(IK) $\qquad \Psi = \{A \in K(\Im): \forall B \in \Im, A \cap B \neq \varnothing\}$.

Let (M, Δ) be a geometric space. If Δ has not the intersection property, and W* is the closure of Δ, the pair (M, W*) is not a simple cooperative game because (W3) is not valid. But we have the following proposition, that generalizes proposition 2.3:

Proposition 3.1 Let (M, Δ) be a finite geometric space and let W be the intersection-closure of Δ. Then (M, W) is a simple cooperative game, called "*the game generated by* (M, Δ)".

Proof. (W1) is evident. If $A \in W$ and $A \subseteq B \subseteq M$, then $\forall C \in \Delta$, $C \cap A \neq \varnothing \Rightarrow C \cap B \neq \varnothing$ and (W2) holds. If $A \in W$ then there exists $C \in \Delta$: $C \subseteq A$ and so $C \cap A^c = \varnothing$ and $A^c \notin W$.

The game (M, W) generated by a geometric space (M, Δ) has two types of blocking coalitions:

(T1) the blocking sets with respect Δ;

(T2) the subsets of M containing at least a block and with intersection void with at least a block.

The losing coalitions are the subsets Y of M non containing blocks and having intersection void with at least a block.

Example 3.1 Let M be a n-set, whose elements are called *players*, and let $\mathfrak{I}$ be a family of subsets non-void of M, called *companies*.
By an economic point of view, we assume that a player belonging to a company has *a power of veto* and a coalition containing a company has *the control* of such company.
Then a winner coalition of the game generated by the geometric space (M, $\mathfrak{I}$) has the control of at least a company and a right of veto on all the companies, a losing coalition don't have a power of veto on at least one company and don't control any company. Finally a blocking coalition of type (T1) has veto for any company but don't control anyone, and a blocking coalition of type (T2) control at least a company but has not veto for all the companies.

We can construct a simple cooperative game by a finite geometric space (M, Δ) also with a "geometric" procedure different from the one of proposition 3.1, by assigning the set Δ^* of minimal winner coalitions.
Precisely, we consider a set Δ^* with the following properties:

(DS1) any $A \in \Delta^*$ is a union of elements of Δ;
(DS2) any element of Δ is contained in at least an element of Δ^*;
(DS3) Δ^* has the intersection and non-inclusion properties;

and we assume W equal to the closure of Δ^*.

We have the following:

Proposition 3.2 Let (M, Δ) be a finite geometric space and let Δ^* be a family of subsets of M satisfying (DS1), (DS2) and (DS3). If W is the closure of Δ^* then:

(DW1) (M, W) is a simple cooperative game;
(DW2) the blocking sets of (M, Δ) are blocking coalitions of (M, W).

Proof. Property (DW1) follows by (DS3). Let X be a blocking set of (M, Δ). Then X and X^c intersect any block and so, by (DS1), any element of Δ^*. By proposition 2.7 it follows that X is a blocking coalition of (M, W).

Proposition 2.5 is a particular case of the proposition 3.2. Another important particular case is concerning the affine planes.

Definition 3.2 A geometric space (M, Δ) is an *affine plane* if:

(AP1) Through any two distinct points there is exactly one line;
(AP2) (*Parallel axiom*) If g is a line and P is a point outside g then there is exactly one line through P that has no points in common with g;
(AP3) There exist three points that are not on a common line.

Let (M, Δ) be a finite affine plane. It is well known that all the lines have the same number $q \geq 2$ of points. The plane is said to be *of order* q and is noted α_q. The number of elements of α_q is q^2 and the lines through a fixed point are q+1.
Let Δ* be a set whose elements are union of two non parallel lines and such that any line of Δ is contained in at least one element of Δ*. We say that Δ* is "*a set of paired lines*". The set Δ* has the intersection and non-inclusion properties and so, by proposition 3.2, we have the following:

Proposition 3.3 Let (M, Δ) be a finite affine plane and let Δ* be a set of paired lines. If W is the closure of Δ* then (M, W) is a simple cooperative game. Moreover, the blocking sets of (M, Δ) are blocking coalitions of (M, W).

In general we can obtain simple cooperative games from *block designs*, in particular from *Steiner systems*.

Definition 3.3 Let t, k, v be natural numbers such that $2 \leq k \leq v$. A finite geometric space (M, Δ) is a *Steiner system* with parameter t, k, v, noted S(t, k, v), if:

(SS1) Through any t distinct points there is exactly one block;
(SS2) Any block has exactly k points;
(SS3) M has v points.

It is well known that necessary conditions for the existence of a S(t, k, v) is the existence of natural positive numbers b_r, $r \in \{0, 1, \ldots, t-1\}$ such that:

$$b_r \binom{k-r}{t-r} = \binom{v-r}{t-r}, \quad r = 0, 1, \ldots, t-1. \tag{3.1}$$

For any $r \in \{0, 1, \ldots, t-1\}$, b_r is the number of blocks through r fixed points. In particular b_o is the number of all the blocks. For t = k the blocks are the subsets of M with k elements and for k = v there is only a block. We say that S(t, k, v) is *non-degenerate* if $t<k<v$.
For t=2 the blocks are called *lines*. If r is a line and P is a point not incident r, $d=b_1-k$ is the number of lines through P non intersecting r. If d = 0, S(2, k, v) is a projective plane and, if d = 1, it is an affine plane.
We have the following:

Proposition 3.4 Let (M, Δ) be a S(2, k, v) and let d = b_1–k. Then:

- k divides d-d^2;
- if r and s are incident lines, the number of lines not incident to r∪s is

$$\alpha = d^2\text{-}d - (d^2\text{-}d)/k. \tag{3.2}$$

Proof. By (3.1) we have:

$$v = (k+d)(k-1) + 1, \qquad b_0 = k^2+(2d-1)k+(d-1)^2+(d-d^2)/k. \tag{3.3}$$

So b_0 is integer if and only if k divides d^2-d. If r and s are two incident lines for each of the k points of r pass k+d-1 lines different from r, for each of the k-1 points of s not belonging to r pass exactly d lines not intersecting r. So the number of lines intersecting r∪s is $\delta = k(k+d-1) + d(k-1) + 1 = k^2 + (2d-1)k - d + 1$
It follows that the lines not intersecting r∪s are $\alpha = b_0 - \delta = d^2\text{-}d + (d\text{-}d^2)/k$.

If d=0 (projective plane) or d=1 (affine plane) we have α=0. Then we assume d>1. Let I[x] be the minimum integer not inferior to x. By previous proposition, we can find a set ρ_{rs} union of at most I[(d^2-d)(k-1)/(2k)] = I[α/2] lines, such that any line of S(2, k, v) intersects r∪s∪ρ_{rs}. Then we have the following:

Proposition 3.5 Let (M, Δ) be a S(2, k, v) with d>1. For any pair (r, s) of incident lines let ρ_{rs} be the union of a minimal set L of lines such that L intersects all the lines not incident to r∪s. Let Δ* be the family of the sets r∪s∪ρ_{rs}, with r, s∈Δ. Then Δ* satisfies (DS1), (DS2) and (DS3) and so generates a set W such that (M, W) is a simple cooperative game. Therefore every element of Δ* is the union of at most I[α/2] + 2 lines.

Example 3.2 For k = 2, a S(2, k, v) is the trivial case of a graph complete with v elements and d = v-3. Any element of Δ* is the union of exactly I[(d^2-d)/4] + 2 lines. For v = k a S(2, k, v) has only a line and d = 0.
Now we consider the non-degenerate Steiner systems with small values of d>1.
For d = 2, by proposition 3.4, is k =2 and so we don't have non-degenerate Steiner systems.
For d = 3, k is a divisor of 6 different from 2. If k = 3 we have a S(2, 3, 13). It is proved (see [17]) that there exists two non isomorphic S(2, 3, 13). In this case we have α = 4 and the elements of Δ* are the union of at most 4 lines. If k = 6 we have a S(2, 6, 46) and α = 5. Then there are at most 5 lines in any element of Δ*.

6. Cooperative games and hyperstructures

In this paragraph we introduce some ideas on the possible relations between cooperative games and some particular commutative weak associative quasi-hypergroups, called "geometric hypergroupoids". We think that it is a very interesting argument of research.

Definition 4.1 Let M be a non-empty set and let $\wp^*(M)$ be the family of non-empty subsets of M. A *hyperoperation* on M is a function $\sigma: M\times M \to \wp^*(M)$, such that to every ordered pair (x, y) of elements of M associates a non-empty subset of M, noted $x\sigma y$. The pair (M, σ) is called *hypergroupoid* with *support* M and *hyperoperation* σ.
If A and B are non-empty subsets of M, we put $A\sigma B = \cup\{a\sigma b: a\in A, b\in B\}$.
Moreover, $\forall a, b\in M$, we put, $a\sigma B = \{a\}\sigma B$ and $A\sigma b = A\sigma\{b\}$.

Definition 4.2 A hypergroupoid (M, σ) is said to be:

(SI) a *semihypergroup*, if $\forall x, y, z\in M$, $x\sigma(y\sigma z) = (x\sigma y)\sigma z$ (*associativity)*;
(QI) a *quasihypergroup*, if $\forall x\in M$, $x\sigma M = M = M\sigma x$ (*riproducibility*);
(HY) a *hypergroup* if it is both a *semihypergroup* and a *quasihypergroup*;
(CO) *commutative*, if $\forall x, y\in M$, $x\sigma y = y\sigma x$;
(WA) *weak associative*, if $\forall x, y, z\in M$, $x\sigma(y\sigma z) \cap (x\sigma y)\sigma z \neq \varnothing$;
(CL) *closed*, if $\forall x, y\in M$, $\{x, y\}\subseteq x\sigma y$;
(IP) *idempotent*, if $\forall x\in M$, $x\sigma x=\{x\}$.

Definition 4.3 We say that a hypergroupoid (M, σ) is *geometric* if it is commutative, closed and idempotent.
A geometric hypergroupoid (M, σ) is said to be a *join space* if the following *incidence axiom* holds:

(IA) $\forall a, b, c, d\in M, (\exists x\in M: a\in b\sigma x, c\in d\sigma x) \Rightarrow (\exists y\in M: y\in a\sigma d\cap b\sigma c)$.

Definition 4.4 Let (M, σ) be a geometric hypergroupoid. A geometric space (M, Δ) is said to be "associated to (M, σ)" if Δ is the set of the hyperproducts $a\sigma b$ with $a\neq b$.

Proposition 4.1 Let (M, Δ) be a space of lines. Then there exists only a geometric hypergroupoid (M, σ) with (M, Δ) associated geometric space. Precisely we have:

(GHA) $\forall x\in M$, $x\sigma x = \{x\}$, $\forall x, y\in M$, $x\neq y$, $x\sigma y$ is the line containing $\{x, y\}$.

Example 4.1 Let M be the support of a projective space and, for any $x, y\in M$, with $x \neq y$, put $x\sigma x = \{x\}$ and $x\sigma y$ equal to the line xy. The hypergroupoid (M, σ) is

geometric and is a *hypergroup*. It is also a *join space*. The incidence axiom is the Veblen-Young axiom.

Example 4.2 Let M be the support of an Euclidean space and, for any x, y∈M, with x ≠ y, put xσx = {x} and xσy equal to the segment xy. The hypergroupoid (M, σ) is *geometric*. It is a *hypergroup* and a *join space*, but not a space of lines, and the incidence axiom is the Pasch axiom.

Example 4.3 Let M be the support of an affine space and, for any x, y∈M, with x ≠ y, put xσx = {x} and xσy equal to the line xy. (M, σ) is a *geometric* hypergroupoid and a *space of lines* but not a hypergroup. The incidence property is not valid.

In the sequel we don't distinguish from a geometric hypergroupoid and the geometric space associated. By previous considerations we have that the concept of *geometric hypergroupoid* generalizes the one of *space of lines* and so projective spaces, affine spaces and Steiner systems S(2, k, v) are particular cases.
The space of lines that are hypergroups or join spaces, e. g. projective spaces, have very interesting properties. Also join spaces that are not spaces of lines such as the one of example 4.2, have important properties.
Let (M, σ) be a geometric hypergroupoid. We call *blocks of order* 1 the singletons {a}, a∈M, *blocks of order* 2 the hyperproducts aσb with a≠b and, for n≥3, we call *blocks of order* n the hyperproducts HσK, with H block of order h<n and K block of order n-h, that are not blocks of order less than n.
A block of order n generalizes the concept of subspace of dimension n-1 of a space of lines and so we can generalize the results of the previous paragraphs. We denote by Δ_n the set of hyperproducts of order n and by Λ_n the set of hyperproducts of order h≤n. Moreover we put $\Lambda_0 = \cup_{n\in N} \Lambda_n$. Then by a geometric hypergroupoid (M, σ) we obtain the geometric spaces (M, Δ_n), with n belonging to a subset of N, finite if M is finite. We have also the geometric spaces (M, Λ_n), $n\in N_0$. In particular $\Delta = \Delta_2$.
Suppose M is a finite set of players. From each of the geometric spaces (M, Δ_n), n>1, we can obtain a cooperative game with particular properties dependent on the algebraic structure of (M, σ).
A possible economic interpretation of a block B of order n is as the set of the players disposed to form a coalition because influenced by the set of players $\{a_1, a_2, \ldots, a_n\}$ that generates the block. If (M, σ) is not a hypergroup such coalition depend on the process of aggregation of the n players. Another possible interpretation of the block B is as a company controlled by $\{a_1, a_2, \ldots, a_n\}$.
Finally, we think that many other economic interpretations and geometric properties (e. g. blocking coalitions) depends on the algebraic structure of (M, σ) and we intend study these questions in a very near paper.

Bibliography

[1] K. J. Arrow, *Social Choose and Individual Values*, J. Wiley and Sons, New York, 1951, 2nd ed 1963.
[2] L. Berardi, F. Eugeni, *Blocking sets in projective plane of order four*, Ann. of Discrete Math. 37 (1988), 43-50.
[3] L. Berardi, F. Eugeni, *Blocking sets e Teoria dei Giochi*, Atti Sem. Mat. Fis. Univ. Modena 34 (1988), 165-196.
[4] L. Berardi, F. Eugeni, *Blocking sets and game theory II*, Proc. "Blocking sets", Gissen, 1989, Springer.
[5] L. Berardi, S. Innamorati, *Irreducible blocking sets in the projective plane of order five*, Atti Sem. Mat. Fis. Univ. Modena, 38 (1990), 293-311.
[6] A. Beutelspacher, *Blocking sets and partial spreads in finite projective spaces*, Geom. Dedicata 9 (1980), 425-449.
[7] A. Beutelspacher, U. Rosenbaum, *Projective Geometry*, Cambridge University Press, 1998
[8] M. Biliotti, G. Korchmaros, *Transitive blocking sets in cyclic projective planes*, Mitt. Math. Sem. Giessen, 201, 35-38.
[9] A. Blokhuis, *Blocking sets in Desarguesian planes*, Proc. Int. Conf. Comb. "Paul Erdos is Eighty", Bolyai Soc. Math. Studies, (1993), 1-20.
[10] A. Blokhuis, A. E. Brower, *Blocking sets in Desarguesian projective planes*, Bull. London Math. Soc. 18 (1986), 132-134.
[11] A. E. Brouwer, H. A. Wilbrink, *Blocking sets in translation planes*, J. Geom. 19 (1982), 200.
[12] A. A. Bruen, *Baer subplanes and blocking sets*, Bull. Amer. Math. Soc. 76 (1970) 342-344.
[13] A. A. Bruen, *Blocking sets in projective planes*, SIAM J. Appl. Math. 21 (1971), 380-392.
[14] A. A. Bruen, J. A. Thas, *Blocking sets*, Geom. Dedicata 6 (1977), 193-203.
[15] E. Burger, *Introduzione alla teoria dei giochi*, Franco Angeli editore, 1967
[16] P.J.Cameron, *Four lectures on finite geometry*, in Finite geometries, editors C. A. Baker, L.M.Batten, Lecture Notes in Pure and Appl. Math. 103, M.Dekker, New York, 1985, 27-63.
[17] M. Cerasoli, F. Eugeni, M. Protasi, Introduzione alla Matematica Discreta, Zanichelli, Bologna, 1988
[18] G. Gambarelli, G. Pederzoli, *Metodi di decisione*, Hoepli, 1992
[19] J. W. P. Hirschfeld, *Projective geometries over finite fields*, Clarendon Press, Oxford, 1979.
[20] T. Hilles, T. Szonyi, F. Wettl, *Blocking sets and maximal strong representative systems in finite projective planes*, Mitt. Math. Sem.Giessen 201 (1991), 97-107.
[21] A. J. Hoffman, M. Richardson, *Block design games*, Canad. J. Math. 13 (1961), 110-128
[22] S. Innamorati, *Sulle coalizioni in teoria dei giochi*, Ratio Math. 2 (1991), 139-150

[23] S. Innamorati, A. Maturo, *On irreducible blocking sets in projective planes*, Ratio Math. 2, (1991), 151-155.
[24] S. Innamorati, A. Maturo, *On blocking sets of smallest cardinality in the projective plane of order seven*, Combinatorics '88, Mediterranean Press, Cosenza, Italy, 1991, 79-96.
[25] S. Innamorati, A. Maturo, *The spectrum of minimal blocking sets*, Discrete Mathematics, 208/209 (1999) 339-347
[26] S. Innamorati, F. Zuanni, *Minimum blocking configurations*, J. Geom. 55 (1996) 86-98.
[27] S. Innamorati, *The non-existence of certain large minimal blocking sets*, Mitt. Math. Sem. Giessen, to appear.
[28] M. Li Calzi, *Teoria dei giochi*, Etas Libri 1995
[29] H. Neumann, *On some finite non-Desarguesian planes*, Arch. Math., 6 (1955), 36-40.
[30] G. Oven, *Game Theory*, Academic Press, New York, 1982
[31] T. G. Ostrom, F. A. Sherk, *Finite projective planes with affine subplanes*, Canad. Math. Bull. 7 (1964), 549-560.
[32] M. Richardson, *On finite projective games*, Proc. Amer. Math. Soc., 7 (1956), 458-465
[33] J. Rosenmuller, *Some topics of cooperative game theory*, in Modern Applied Math., North Holland, 1982
[34] L. S. Shapley, *Simple Games - An outline of the theory*, Rand Corporation P-series Report, 1962
[35] G. P. Szego, G. Gambarelli, *Discontinuos solutions in n-persons Games*, in "New quantitative techniques for Economic Analisys". Accademic Press, 1982
[36] J. Singer, *A theorem in finite geometry and some application to number theory*, Trans. Amer. Math. Soc. 43 (1938), 377-385.
[37] T. Szonyi, *Note on existence of large minimal blocking sets in Galois planes*, Combinatorica 12 (1992), 227-235.
[38] G. Tallini, *On blocking sets in finite projective and affine spaces*, Ann. of Discrete Math. 37 (1988), 433-450.
[39] G. Tallini, *Geometrie di incidenza e matroidi*, IAC Roma, Quad. Serie III-N.127 (1981), 433-450.
[40] J. Von Neumann, O. Morgenstern, *Theory of games and economic behaviour*, Princeton, 1947

Chemical Examples in Hypergroups

B. Davvaz[1] and A. Dehghan-Nezhad[1]

Abstract Hypergroups first were introduced by Marty in 1934. Up to now many researchers have been working on this field of modern algebra and developed it. It is purpose of this paper to provide examples of hypergroups associated with chemistry. The examples presented are connected to construction from chain reactions.

AMS Subject Classification: 20N20

Keywords and Phrases: hyperstructure, hypergroups, chain relation.

1 Introduction

The theory of algebraic hyperstructures which is a generalization of the concept of algebraic structures first was introduced by Marty in 1934 [4], and had been studied in the following decades and nowadays by many mathematicians, and many papers concerning various hyperstructures have appeared in the literature, for example see [2,3,6,8]. The basic definitions of the object can be found in [1,7].

Definition. A hyperstructure is a non-empty set S together with a function $\cdot : S \times S \longrightarrow P^*(S)$ called hyperoperation, where $P^*(S\}$ denotes the set of all non-empty subsets of S. If $A, B \subseteq S$, $x \in S$ then we define

$$A \cdot B = \bigcup_{a \in A, b \in B} a \cdot b, \quad x \cdot B = \{x\} \cdot B, \quad \text{and } A \cdot x = A \cdot \{x\}.$$

The hyperoperation $\cdot$ is called associative in S if

$$(x \cdot y) \cdot z = x \cdot (y \cdot z) \text{ for all } x,y,z \text{ in } S.$$

Definition. A hyperstructure $(S , \cdot)$ is called a hypergroup [1] if

i) $(\cdot)$ is associative.

ii) $a \cdot S = S \cdot a = S$ for all $a \in S$.

Definition. A non-empty subset K of the hypergroup S is called a subhypergroup of S if $a \cdot K = K \cdot a = K$ for all $a \in K$.

[1] Department of Mathematics, University of Yazd, Yazd, Iran
E-mail: *davvaz@yazduni.ac.ir*

In this paper, we will give some examples of hypergroups associated with chemistry. The examples presented are connected to construction from chian reactions.

2 Preliminaries

a) Chain reactions
An atom of group of atoms possessing an odd (unpaired) electron is called a free radical, such as

$$Cl,\ CH_3,\ C_2H_5$$

The chlorination of methane is an example of a chain reaction, a reaction that involves a series of steps, each of which generates a reactive substance that brings about the next step. While chain reactions may vary widely in their details, they all have certain fundamental characteristics in common.

1) $Cl_2 \longrightarrow 2Cl^o$
(1) is called Chain-initiating step.

2) $Cl^o + CH_4 \longrightarrow HCl + CH_3{}^o$

3) $CH_3{}^o + Cl_2 \longrightarrow CH_3Cl + Cl^o$
then (2), (3), (2), (3), etc, until finally:
(2) and (3) are called Chain-propagating steps.

4) $Cl^o + Cl^o \longrightarrow Cl_2$ or

5) $CH_3{}^o + CH_3{}^o \longrightarrow CH_3CH_3$ or

6) $CH_3{}^o + Cl^o \longrightarrow CH_3Cl.$
(4),(5) and (6) are called Chain-terminating steps.

First in the chain of reactions is a chain-initiating step, in which energy is absorbed and a reactive particle generated; in the present reaction it is the cleavage of chlorine into atoms (step 1).
There are one or more chain-propagating steps, each of which consumes a reactive particle and generates another; there they are the reaction of chlorine atoms with methane (step 2), and of methyl radicals with chlorine (step 3).

Finally, there are chain-terminating steps, in which reactive particles are consumed but not generated; in the chlorination of methane these would involve the union of two of the reactive particles, or the capture of one of them by the walls of the reaction vessel.

b) The Halogens *F, Cl, Br,* and *I*
The halogens are all typical non-metals. Although their physical forms differ-fluorine and chlorine are gases, bromine is a liquid and iodine is a solid at room temperature, each consists of diatomic molecules; F_2, Cl_2, Br_2 and I_2. The halogens

all react with hydrogen to form gaseous compounds, with the formulas *HF*, *HCl*, *HBr*, and *HI* all of which are very soluble in water. The halogens all react with metals to give halides.

$$:\ddot{F}-\ddot{F}:, \qquad :\ddot{Cl}-\ddot{Cl}:, \qquad :\ddot{Br}-\ddot{Br}:, \qquad :\ddot{I}-\ddot{I}:$$

The reader will find in [5] a deep discussion of chain reactions and halogens.

3 Chemical Hypergroups

In during chain reaction

$$A_2 + B_2 \xleftrightarrow{\text{Heat or Light}} 2AB$$

there exist all molecules A_2, B_2, AB and whose fragment parts A^o, B^o in experiment. Elements of this collection can by combine with each other.

All combinational probability for the set $S = \{ A^o, B^o, A_2, B_2, AB\}$ to do without energy can be displayed as follows:

+	A^o	B^o	A_2	B_2	AB
A^o	A^o,A_2	A^o,B^o,AB	A^o,A_2	A^o,B_2,B^o,AB	A^o,AB,A_2,B^o
B^o	A^o, B^o,AB	B^o, B_2	A^o,B^o,AB, A_2	B^o,B_2	A^o, B^o,AB, B_2
A_2	A^o, A_2	A^o, B^o,AB,A_2	A^o, A_2	A^o, B^o,A_2,B_2,AB	A^o, B^o, A_2,AB
B_2	A^o,B^o,B_2,AB	B^o,B_2	A^o, B^o,A_2, B_2,AB	B^o, B_2	A^o, B^o, B_2,AB
AB	A^o,AB,A_2,B^o	A^o,B^o,AB,B_2	A^o, B^o,A_2,AB	A^o, B^o, B_2,AB	A^o, B^o, A_2, B_2,AB

Theorem. (S , +) is a hypergroup.

Proof. Clearly reproduction axiom and associativity are valid. As a sample of how to calculate the associativity, we illustrate some cases:

$$\begin{cases} (AB+A_2)+B_2 = \{ AB, A_2, A^o, B^o \}+B_2 = \{B_2, AB, A_2, A^o, B^o \}, \\ AB+(A_2+B_2)=AB+\{A_2, B_2, A^o, B^o, AB \} = \{ A_2, B_2, AB, A^o, B^o\}, \end{cases}$$

$$\begin{cases} (AB+A^o)+A^o=\{ AB, A^o, A_2, B^o \}+A^o = \{A_2, A^o, AB, B^o \}, \\ AB+(A^o+A^o)=AB+ \{A_2, A^o \} = \{ A_2, AB, A^o, B^o\}, \end{cases}$$

$$\begin{cases} (A_2+B^o)+B_2 = \{ AB, A^o, A_2, B^o \}+B_2 = \{B_2, AB, B^o, A^o, A_2 \}, \\ A_2+(B^o+B_2)=A_2+\{B_2, B^o \} = \{ A_2, A^o, AB, B^o, B_2\}. \end{cases}$$

Corollary. $S_1=\{A^o,A_2\ \}$ and $S_2=\{B^o,B_2\ \}$ are only subhypergroups of $(S\ ,+)$.

If we consider A=H and B $\in$ { F, Cl, Br, I } (for example B = I), the complete reaction table becomes:

+	H^o	I^o	H_2	I_2	HI
H^o	H^o,H_2	H^o,I^o,HI	H^o,H_2	H^o,I_2,I^o,HI	H^o,HI,H_2,I^o
I^o	H^o, I^o,HI	I^o, I_2	H^o,I^o,HI, H_2	I^o,I_2	H^o, I^o,HI, I_2
H_2	H^o, H_2	H^o, I^o,HI,I_2	H^o, H_2	H^o, I^o,H_2,I_2,HI	H^o, I^o, H_2,HI
I_2	H^o,I^o,I_2,HI	H^o,I_2	H^o, I^o,H_2, I_2,HI	H^o, I_2	H^o, I^o, I_2,HI
HI	H^o,HI,H_2,I^o	H^o,I^o,HI,I_2	H^o, I^o,H_2,HI	H^o, I^o, H_2,HI	H^o, I^o, H_2, I_2,HI

Acknowledgment
We appreciate the assistance and suggestions of Dr. A. Gorgi at the Department of Chemistry.

References

[1] P. Corsini, *Prolegomena of hypergroup theory*, second edition, Aviani editor, (1993).
[2] M.R. Darafsheh and B. Davvaz, *H_v-ring of fractions*, Italian J. Pure Appl. Math., 5 (1999) 25-34.
[3] B. Davvaz, *Weak Polygroups*, Proc. 28th Annual Iranian Math. Conf, (1977), 139-145.
[4] F. Marty, *Sur une generalization de la notion de groupe*, 8^{iem} congres Math. Scandinaves, Stockhlom, (1934), 45-49.
[5] Morrison and Boyd, *Organic Chemistry*, Sixth Eddition, Prentice-Hall, Inc, 1992.
[6] T. Vougiouklis, *A new class of hyperstructures*, J. Combin. Inf. system Sci, 20, (1995), 229-235.
[7] T. Vougiouklis, *Hyperstructures and their repesentations*, Hadronic Press Inc, (1994).
[8] T. Vougiouklis, *Convolution on WASS hyperstructures*, Discrete Math., (1997), 347-355.

AN EXAMPLE OF A JOIN SPACE ASSOCIATED WITH A RELATION

Laurenţiu LEOREANU[1]

Abstract In this paper a join spaces associated with a binary relation is presented.

Keywords Join spaces, Relations

First of all, let us recall what a join space is.
Let H be a nonemptyset and o : H x H $\rightarrow P^*(H)$, where $P^*(H)$ is the set of nonempty subsets of H.
If $A \subset H$, $B \subset H$, then we set $A \circ B = \cup_{a\in A} \cup_{b\in B} a \circ b$.
We denote $A \approx B$ if $A \cap B \neq \emptyset$.
If the hyperoperation "o" is associative and $\forall a \in H$, we have $a \circ H = H = H \circ a$, then (H,o) is a **hypergroup.**
Denote $a / b = \{x \in H \mid a \in b \circ x\}$, for any $(a,b) \in H^2$.
A hypergroup (H,o) is called a join space if "o" is commutative and

$$a / b \approx c / d \Rightarrow a \circ d \approx b \circ c.$$

Join spaces have been introduced by W. Prenowitz and used by himself and J. Jantosciak in order to rebuild some branches of non-Euclidian geometries. Afterwards, join spaces have also been used in the study of other topics (Graphs and Hypergraphs, Lattices, Binary Relations and so on).
Here, a connection between join spaces and reflexive and symmetric relations is presented.
First, we give an example:
Let $f : H \rightarrow U$ be an onto map.
We define on H the following hyperoperation:

$$\forall (x,y) \in H^2,\ x \circ x = f^{-1}(f(x)),\ x \circ y = x \circ x \cup y \circ y$$

(where $\forall Y \subset U$, $f^{-1}(Y) = \{x \in H \mid f(x) \in Y\}$).

Proposition (H,o) is a join space.

Proof. For any $(x,y,z) \in H^3$, we have
$(x \circ y) \circ z = x \circ (y \circ z) = f^{-1}(f(x)) \cup f^{-1}(f(y)) \cup f^{-1}(f(z))$
and $x \circ H = \cup_{a\in H} x \circ a = \cup_{a\in H} f^{-1}(f(x)) \cup f^{-1}(f(a)) = H$,
since f is onto.
So, (H, o) is a commutative hypergroup.

[1] Grupul Şcolar "Miron Costin " Roman, România

Moreover, any $x \in H$ is an identity of H (since $\forall\, y \in H,\ \ y \in x \circ y$) and for any $(x,y) \in H^2$, x is an inverse of y.
Let us check now that $a / b \approx c / d \Rightarrow a \circ d \approx b \circ c$.
Let $x \in a / b \cap c / d$ that is $a \in f^{-1}(f(x)) \cup f^{-1}(f(y))$ and $c \in f^{-1}(f(x)) \cup f^{-1}(f(d))$.
It follows $f(a) \in \{f(x), f(b)\}$ and $f(c) \in \{f(x), f(d)\}$.
We must prove that there is $y \in H$, such that $y \in a \circ d \cap b \circ c$, that is
$f(y) \in \{f(a), f(d)\} \cap \{f(b), f(c)\}$.
We have the following situations:

1) if $f(a) = f(x) = f(c)$ then we can choose $y = a$;
2) if $f(a) = f(x)$ and $f(c) = f(d)$, then we choose $y = d$;
3) if $f(a) = f(b)$, then we choose $y = a$.

Therefore, (H, o) is a join space.

Now, let us consider R a reflexive and symmetric relation on H.
Let us consider the following hyperoperation on H
$\forall (x,y) \in H^2$, $x \circ_R x = \{z \mid (z,x) \in R\}$, $x \circ_R y = x \circ_R x \cup y \circ_R y$.

Theorem $(H, \circ_R)$ is a join space.

Proof. The associativity is immediate and $\forall\, x \in H$, we have
$x \circ_R H = x \circ_R x \cup \cup_{a \in H}\, a \circ_R a = H$, since
$\cup_{a \in H}\, a \circ_R a = \cup_{a \in H} \{z \mid (z,a) \in R\} = H$ (R is reflexive).
So, $(H, \circ_R)$ is a commutative hypergroup.
Notice that $a \in a \circ_R a \Leftrightarrow (a,a) \in R$.
Let us check now that $a / b \approx c / d \Rightarrow a \in a \circ_R d \approx b \circ_R c$.
Let $x \in H$, such that $a \in x \circ_R b$ and $c \in x \circ_R d$.
We have $a \in \{t \mid (t,x) \in R$ or $(t,b) \in R\}$, whence $(a,x) \in R$ or $(a,b) \in R$.
Similarly, $(c,x) \in R$ or $(c,d) \in R$.
We must prove that there is $y \in H$, such that $y \in a \circ_R d$ and $y \in b \circ_R c$, that is
$[(y,a) \in R$ or $(y,d) \in R]$ and $[(y,b) \in R$ or $(y,c) \in R]$,
or equivalently, $[(y,a) \in R$ and $(y,b) \in R]$ or $[(y,a) \in R$ and $(y,c) \in R]$
or $[(y,d) \in R$ and $(y,b) \in R]$ or $[(y,d) \in R$ and $(y,c) \in R]$.
We have the following situations:

1) $(a,x) \in R$ and $(c,x) \in R$. Since R is symmetric, it follows $(x,a) \in R$ and $(x,c) \in R$. In this case, we can choose $y=x$.
2) $(a.x) \in R$ and $(c,d) \in R$. We take $y=c$ and so, $(y,d)=(c,d) \in R$ and $(y,c)=(c,c) \in R$.
3) $(a,b) \in R$ and $[(c,x) \in R$ or $(c,d) \in R]$.
4) We take $y=a$, so $(y,b)=(a,b) \in R$ and $(y,a)=(a,a) \in R$.

Therefore, $(H, \circ_R)$ is a join space.

Remark. If R is the relation defined as follows:
$(x,y) \in R \Leftrightarrow f(x)=f(y)$
where $f : H \to U$, then (H, o_R) is the join space presented at the beginning.

References

[1] Chvalina, J., Relational product of join spaces determined by quasi-orders, Sixth.Int.Congress on AHA(1996), 15-23, Democritus University of Thrace Press.
[2] Corsini, P., Prolegomena of Hypergroup theory, Aviani Editore, 1993.
[3] Corsini, P., On the hypergroups associated with binary relations, J. Multiple-valued Logic, 2000, vol.5, p.407-419.
[4] Corsini, P., Binary relations and hypergroupoids, Italian J. of Pure and Appl. Math., n. 7, 2002.
[5] Corsini, P., Leoreanu, V., Hypergroups and binary relations, Algebra Universalis, 43, 2000, p. 321-330.
[6] Corsini, P., Leoreanu, V., Applications of Hyperstructure Theory, Kluwer Academic Publishers, 2002.
[7] Leoreanu, V., Leoreanu, L., Hypergroups associated with hypergraphs, Italian J. of Pure and Applied Mathematics, 4, 1998, p. 119-126.
[8] Rosenberg, I.G., Hypergroups and join spaces determined by relations, Italian Journal of Pure and Applied Math., nr. 4, 1998, p. 93-101.