

AD 673220A



CONVAIN
ASTRONAUTICS
DIVISION
LIBRARY



CONVAIN (ASTRONAUTICS) DIVISION
GENERAL DYNAMICS CORPORATION

a

DDC
REF
JUL 24 1968
LIBRARY



REPORT NO. AG-589
DATE 7 Oct. 1959
NO. OF PAGES 5

CONVAIR | ASTRONAUTICS

CONVAIR DIVISION OF GENERAL DYNAMICS CORPORATION

Distribution of this document
is unlimited.

AD672204

THE WEAPONS-ASSIGNMENT PROBLEM



PREPARED BY E. W. Cheney
E. W. Cheney

CHECKED BY _____

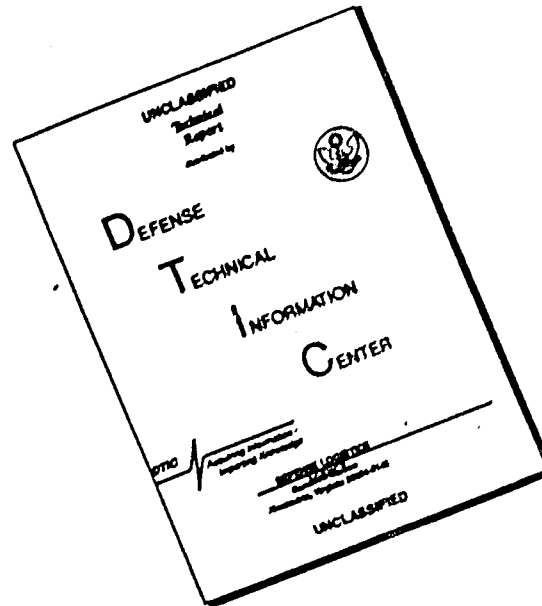
APPROVED BY _____

APPROVED BY _____

REVISIONS

NO.	DATE	BY	CHANGE	PAGES AFFECTED

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

The problem of the title is as follows. There is a stockpile of n missiles and a list of m military targets. The value of the i^{th} target is a known quantity V_i ($1 \leq i \leq m$). The probability that the i^{th} target will not be hit if attacked by the j^{th} missile is a known quantity P_{ij} ($1 \leq i \leq m$, $1 \leq j \leq n$). It is assumed that a hit results in total destruction of the target. What assignment of missiles to targets will effect the maximum expected destruction?

If all missiles were directed to the same target, say the i^{th} , the destruction expected would be

$$D = V_i \left[1 - \prod_{j=1}^n P_{ij} \right].$$

This formula is trivial for $n = 1$, and may be proved inductively for other values of n as follows: A strike from a salvo of n missiles can result in three mutually exclusive events:

- (a) one of the first $(n-1)$ missiles strikes while the n^{th} misses,
- (b) one of the first $(n-1)$ missiles strikes and the n^{th} strikes also,
- (c) none of the first $(n-1)$ missiles strikes while the n^{th} strikes,

The probabilities of these three events being added, we obtain, assuming as an induction hypothesis that the formula for D is valid up to $n-1$:

$$\left(1 - \prod_{j=1}^{n-1} P_{ij} \right) P_{in} + \left(1 - \prod_{j=1}^{n-1} P_{ij} \right) \cdot (1 - P_{in}) + \left(\prod_{j=1}^{n-1} P_{ij} \right) (1 - P_{in}).$$

Algebraically, this is equivalent to $1 - \prod_{j=1}^n P_{ij}$, Q.E.D.

Now consider the given data as being laid out tabularly.

V_1	P_{11}	P_{12}	...	P_{1n}
V_2	P_{21}	P_{22}	...	P_{2n}
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
V_n	P_{n1}	P_{n2}	...	P_{nn}

An assignment of weapons to targets can be represented by the following table. In this table to each row there is one target and to each column there is one weapon. The numbers in the table indicate the number of weapons assigned to each target. For example, a five-target, three-weapon assignment is shown in the table below. The numbers indicate that one weapon is assigned to target 1, two weapons are assigned to target 2, and one weapon is assigned to target 3.

1	2	1	0	0
0	2	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

The assignment of weapons to targets can be represented by the following table.

1 2 3 4 5

The assignment of weapons to targets can be represented by the following table.

In the next paragraph there is described a relaxation method for the improvement of assignments. This algorithm does not generally yield a solution of the original problem, but is offered here as a means of generating good if not optimal assignments by means of automatic computation. This method, in fact, solves the following simpler problem: as missiles roll off the assembly line, it is desired to assign them optimally and sequentially to targets without altering the assignments of those already stockpiled. There may be instances where the problem just described is the more practical, as when altering the assignment of a missile involves costly reinstrumentation.

Now suppose, for the moment, that an assignment has been fixed for all missiles except one, say the r^{th} one. To which target should the r^{th} missile be directed in order that the resulting full assignment shall be optimal? This problem can be solved by calculating the m different total destructions that can result when missile r is directed to targets $1, 2, \dots, m$ in turn. Having done this for one value of r , we repeat for another, at each step changing one component of the assignment vector to increase the total destruction. This process, repeated indefinitely, is called relaxation.

If all but the r^{th} missile are fired, the probability of survival of target i is

$$\prod_{\substack{j=1 \\ j \neq r}}^n H_{ij}.$$

The expected (surviving) value of target i is then

$$V_i \prod_{\substack{j=1 \\ j \neq r}}^n H_{ij}.$$

If missile r is fired at target i , an additional expected destruction is effected of magnitude

$$E_i = (1 - P_{ir}) V_i \prod_{j=1}^n H_{ij}.$$

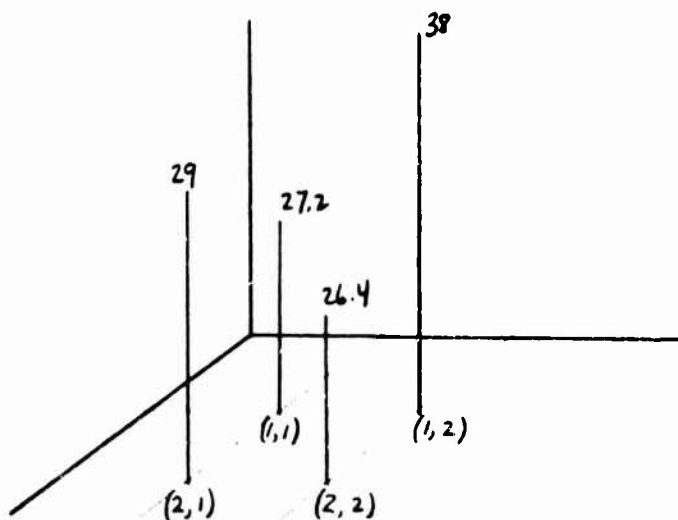
Missile r should then be assigned to that target i for which E_i is a maximum. A program for the IBM 650 which performs this task iteratively is described in Appendix I. We show now by example that the process can fail to produce a solution. Consider the datum

V	P	
40	.5	.8
30	.3	.4

The assignment (2,1) yields a total expected destruction of

$$D = (.7)(30) - (.2)(40) = 29.$$

Any attempt to improve this by alteration of a single component will fail, for $D(1,1) = 27.2$ and $D(2,2) = 26.4$. However, $D(1,2) = 38$. The assignment (1,2) can be obtained from (2,1) only by alteration of two components simultaneously. The example just considered may be represented pictorially as follows. The assignments (1,1), (1,2), (2,1), (2,2) may be shown as points in E_2 , and the corresponding value of D may be plotted on the vertical axis.



The relaxation algorithm can proceed from (2,2) to (1,2), or from (1,1) to (1,2), but not from (2,1) to (1,2).

Another example of the same type is as follows:

V	P		
10	.4	.5	.5
20	.8	.7	.5
10	.5	.5	.1

The assignment (2,1,3) yields an expected destruction of

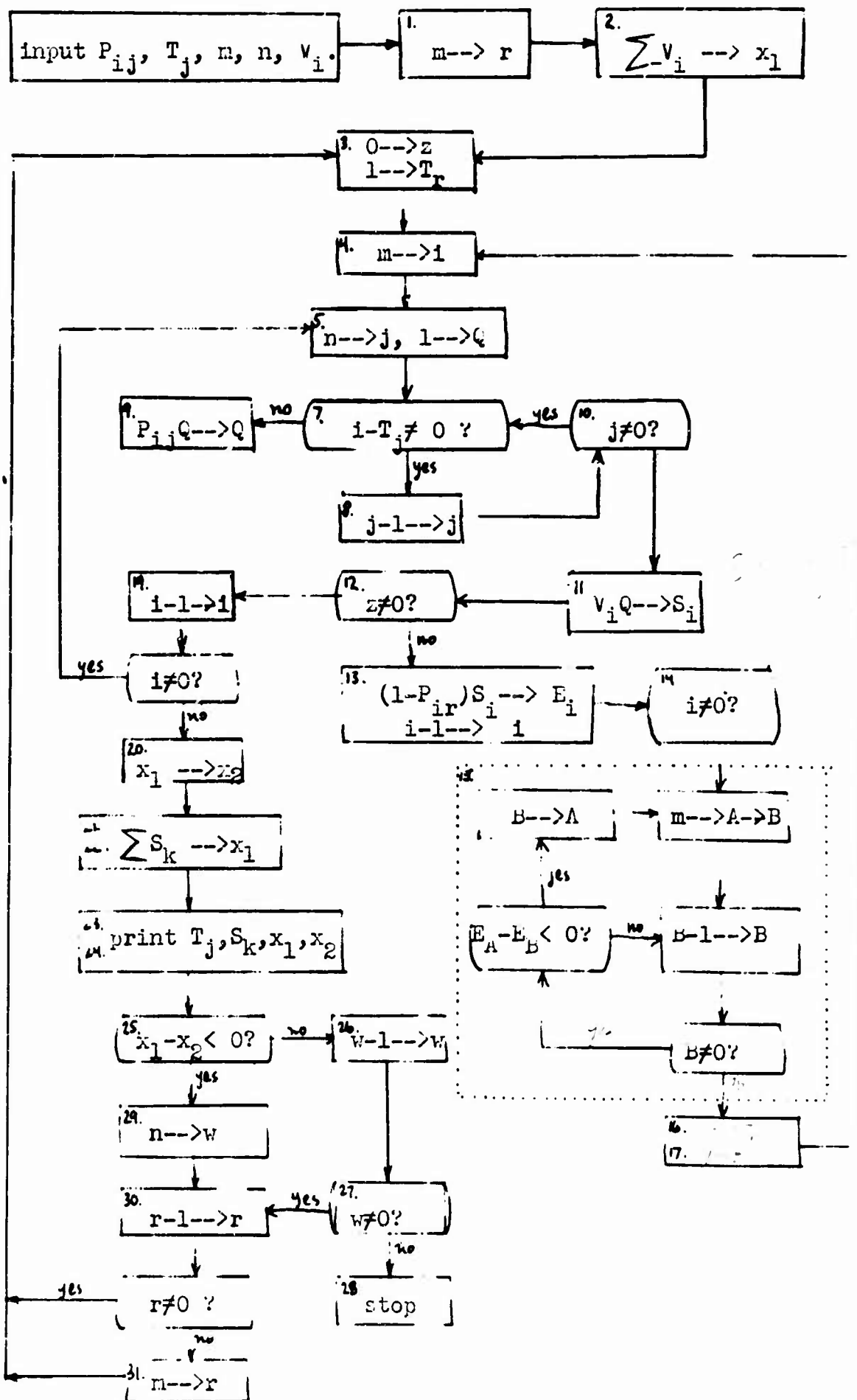
$$D = 20(.2) + 10(.5) + 10(.9) = 18.$$

Every assignment that can be obtained from (2,1,3) by altering a single component yields a lower value of D . However, there exists an assignment, viz. (1,2,3), obtainable from (2,1,3) by altering two components, which yields a higher value of D .

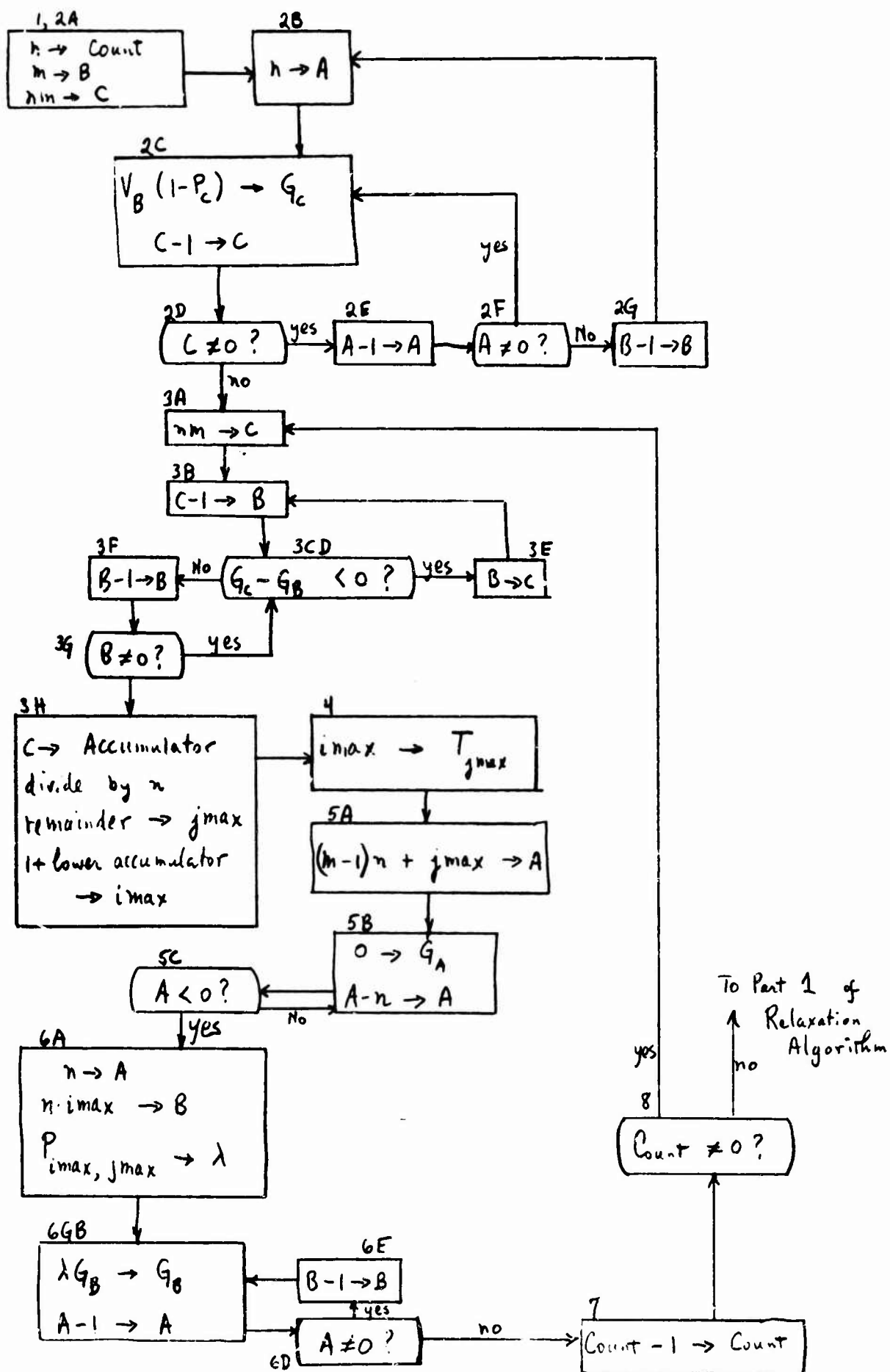
In an attempt to improve an assignment produced by the relaxation method, we may try interchanges of components in the assignment vector T . Both the preceding examples would yield to this treatment. It is not known whether these two methods together will produce optimum assignments in all cases. It seems unlikely that they would. This composite algorithm may be tried over and over on different initial assignments, and these initial assignments may be generated by a random device in the computer. Measures such as this are of use even in simpler problems like the classical assignment problem. In using the relaxation algorithm, it is possible to start with any initial assignment. A reasonable first assignment may be obtained as follows: Calculate all numbers $V_i (1 - P_{ij})$. Let the greatest be indexed with i_0, j_0 . Then, in the initial assignment, send missile j_0 to target i_0 . Now reduce V_{i_0} to $V_{i_0} P_{i_0 j_0}$ and repeat the process, omitting j_0 as a value of j . The 650 Program which accomplishes this is described in Appendix II.

All the measures just described must be regarded only as stopgaps until a genuine solution becomes available.

Appendix I: Relaxation Algorithm



Appendix II: Starting Procedure for Relaxation Algorithm



Line	Label	Instruction	1950	1999	MISSILE ASSIGNMENT PROBLEM
1		BLR	1950	1999	
2		REG	P0101	0600	
3		REG	V0601	0630	
4		REG	T0631	0650	
5		REG	S0651	0680	
6		REG	E0681	0710	
7		REG	Q1027	1036	
8		REG	X1077	1086	
9		REG	Y1127	1136	
10		REG	R1951	1960	
11		SYN	X1	X0001	
12		SYN	X2	X0002	
13		SYN	Z	1027	
14		SYN	START	1950	
15		SYN	HALT	0000	
16		SYN	M	1951	
17		SYN	N	1952	
18		REG	G1200	1800	
19	START	RCD	R0001	1A	1950 70 1951 0001
20	1A	LDD	N		ALGORITHM 0001 69 1952 0005
21		STD	COUNT	2A	TO OBTAIN 0005 24 0008 0011
22	2A	LDD	M		STARTING 0011 69 1951 0004
23		RAB	8001		ASSIGNMENT 0004 82 8001 0010
24		RAU	8001		FOR 0010 60 8001 0017
25		MPY	N		RELAXATION 0017 19 1952 0022
26		RAC	8002	2B	METHOD 0022 88 8002 0031
27	2B	LDD	N		0031 69 1952 0055
28		RAA	8001	2C	0055 80 8001 0061
29	2C	RAU	FLONE		0061 60 0014 0019
30		FSB	P0000	C	0019 33 6100 0027
31		FMP	V0000	B	0027 39 4600 0050
32		STU	G0000	C	0050 21 7199 0002
33		SXC	0001	2D	0002 59 0001 0058
34	2D	NZC	2E	3A	0058 48 0711 0012
35	2E	SXA	0001	2F	0711 51 0001 0067
36	2F	NZA	2C	2G	0067 40 0061 0021
37	2G	SXB	0001	2B	0021 53 0001 0031
38	3A	RAU	N		0012 60 1952 0007
39		MPY	M		0007 19 1951 0072
40		RAC	8002	3B	0072 88 8002 0081
41	3B	RAB	6000		0081 82 6000 0038
42		SXB	0001	3C	0038 53 0001 0044
43	3C	RAU	G0000	C	0044 60 7199 0003
44		FSB	G0000	B	0003 33 5199 0025
45	3D	BMI	3E	3F	0025 46 0028 0029
46	3E	LDD	8006		0028 69 8006 0034
47		RAC	8001	3B	0034 88 8001 0081
48	3F	SXB	0001	3G	0029 53 0001 0035
49	3G	NZB	3C	3H	0035 42 0044 0039
50	3H	RAL	8007		0039 65 8007 0047
51		DIV	N		0047 14 1952 0062
52		STU	JMAX		0062 21 0016 0069
53		ALO	ONE		0069 15 0722 0077
54		STL	IMAX	4A	0077 20 0731 0084
55	4A	LDD	JMAX		0084 69 0016 0719
56		RAA	8001		0719 80 8001 0727
57		LDD	IMAX		0727 69 0731 0734
58		STD	T0000	A 5A	0734 24 2630 0033
59	5A	RSU	ONE		0033 61 0722 0777
60		AUP	M		0777 10 1951 0755
61		MPY	N		G755 19 1952 0772

62		ALO	JMAX		0772	15	0016	0071
63		RAA	8003		0071	80	8003	0030
64		RAU	ZERO	5B	0030	60	0083	0037
65	5B	STU	G0000	A	0037	21	3199	0052
66		LDD	N		0052	69	1952	0805
67		SXA	8001	5C	0805	51	8001	0761
68	5C	BMA	6A	5B	0761	41	0064	0037
69	6A	LDD	N		0064	69	1952	0855
70		RAA	8001		0855	80	8001	0811
71		RAU	N		0811	60	1952	0057
72		MPY	IMAX		0057	19	0731	0752
73		RAB	8002		0752	82	8002	0861
74		SLO	N		0861	16	1052	0757
75		ALO	JMAX		0757	15	0016	0721
76		RAC	8002		0721	88	8002	0079
77		LDD	P0000	C	0079	69	6100	0053
78		STD	LAMBD	6B	0053	24	0006	0009
79	6B	RAU	G0000	B	0009	60	5199	0753
80		FMP	LAMBD		0753	39	0006	0056
81		STU	G0000	B 6C	0056	21	5199	0802
82	6C	SXA	0001	6D	0802	51	0001	0758
83	6D	NZA	6E	7A	0758	40	0911	0712
84	6E	SXB	0001	6B	0911	53	0001	0009
95	7A	RAU	COUNT		0712	60	0008	0013
86		SUP	ONE		0013	11	0722	0827
87		STU	COUNT		0827	21	0008	0961
88		NZU	3A	PT1	0961	44	0012	0066
89	PT1	LDD	N		0066	69	1952	0905
90		STD	R		0905	24	0808	1011
91		LDD	M		1011	69	1951	0054
92		RAA	8001		0054	80	8001	0060
93		SUP	8003	PT2	0060	11	8003	0717
94	PT2	FAD	V0000	A	0717	32	2670	0877
95		SXA	0001		0877	51	0001	0733
96		NZA	PT2		0733	40	0717	0087
97		STU	X1	PT3	0087	21	1077	0080
98	PT3	LDD	ZERO		0080	69	0083	0036
99		STD	Z	PT32	0036	24	1027	0715
100	PT4	LDD	M		0730	69	1951	0754
101		STD	I	PT5	0754	24	0807	0760
102	PT5	LDD	N		0760	69	1952	0955
103		RAA	8001		0955	80	8001	1061
104		RAU	FLONE		1061	60	0014	0769
105		STU	Q	PT7	0769	21	0024	0927
106	PT7	RSU	T0000	A	0927	61	2630	0085
107		AUP	I		0085	10	0807	1111
108		NZU	PT8	PT9	1111	44	0015	0716
109	PT8	SXA	0001	PT10	0015	51	0001	0771
110	PT9	RAU	I		0716	60	0807	1161
111		SUP	ONE		1161	11	0722	0977
112		MPY	N		0977	19	1952	0822
113		ALO	8005		0822	15	8005	0729
114		RAC	8002		0729	88	8002	0737
115		RAU	Q		0737	60	0024	0779
116		FMP	P0000	C	0779	39	6100	0100
117		STU	Q	PT8	0100	21	0024	0015
118	PT10	NZA	PT7	PT11	0771	40	0927	0075
119	PT11	RAU	I		0075	60	0807	1811
120		RAB	8003		1811	82	8003	0020
121		RAU	Q		0020	60	0024	0829
122		FMP	V0000	B	0829	39	4600	0750

RELAXATION
ALGORITHM
BEGINS
HERE

123		STU	S0000 B	PT12	0750	21	4650	0803
124	PT12	RAU	Z		0803	60	1027	0781
125		NZU	PT19	PT13	0781	44	0735	0086
126	PT13	RAU	I		0086	60	0807	1861
127		SUP	CNE		1861	11	0722	1177
128		STU	I		1177	21	0807	0810
129		MPY	N		0810	19	1952	0872
130		ALO	R		0872	15	0808	0063
131		RAA	8002		0063	80	8002	0821
132		RSU	P0000 A		0821	61	2100	1005
133		FAD	FLONE		1005	32	0014	0041
134		FMP	S0000 B		0041	39	4650	0800
135		STU	E0000 B	PT14	0800	21	4680	0783
136	PT14	RAU	I		0783	60	0807	1911
137		NZU	PT5	PT15	1911	44	0760	0766
138	PT15	LDD	M		0766	69	1951	0804
139		RAA	8001		0804	80	8001	0860
140		RAC	8001	BOX1	0860	88	8001	0816
141	BOX1	SXC	0001		0816	59	0001	0922
142		NZC	BOX2	BOX3	0922	48	0725	0026
143	BOX3	LDD	8005		0026	69	8005	0032
144		STD	K	PT16	0032	24	0785	0088
145	BOX2	RAU	E0000 A		0725	60	2680	0835
146		FSB	E0000 C		0835	33	6680	0857
147		BMI	BOX4	BOX1	0257	46	0910	0816
148	BOX4	RAA	6000	BOX1	0910	80	6000	0816
149	PT16	LDD	R		0088	69	0808	0762
150		RAA	8001		0762	80	8001	0018
151		LDD	K		0018	69	0785	0738
152		STD	T0000 A	PT17	0738	24	2630	0833
153	PT17	LDD	ONE		0833	69	0722	0775
154		STD	Z	PT4	0775	24	1027	0730
155	PT19	SXB	0001		0735	53	0001	0091
156		RAU	I		0091	60	0807	0812
157		SUP	ONE		0812	11	0722	1827
158		STU	I		1827	21	0807	0960
159		NZB	PT5	PT20	0960	42	0760	0714
160	PT20	LDD	X1		0714	69	1077	0780
161		STD	X2		0780	24	1078	0831
162		LDD	M		0831	69	1951	0854
163		RAA	8001		0854	80	8001	1010
164		SUP	8003	PT21	1010	11	8003	0767
165	PT21	FAD	S0000 A		0767	32	2650	1877
166		SXA	0001		1877	51	0001	0883
167		NZA	PT21	PT22	0883	40	0767	0787
168	PT22	STU	X1	PT23	0787	21	1077	0830
169	PT23	PCH	X0001	PT24	0830	71	1077	1927
170	PT24	LDD	M		1927	69	1951	0904
171		RAB	8001		0904	82	8001	1060
172		RSC	0000	SET 1	1060	89	0000	0866
173	SET 1	RSA	0008	LOOP1	0866	81	0008	0972
174	LOOP1	LDD	S0001 C		0972	69	6651	0954
175		STD	Y0009 A		0954	24	3135	0788
176		AXC	0001		0788	58	0001	0094
177		SXB	0001		0094	53	0001	0850
178		NZB		THRU1	0850	42	0853	1004
179		AXA	0001		0853	50	0001	0059
180		NZA	LOOP1		0059	40	0972	0713
181		PCH	Y0001		0713	71	1127	0078
182		RSA	0008		0078	81	0008	0784
183		LDD	ZERO	LOOP2	0784	69	0083	0736

184	LOOP2	STD	Y0009	A	0736	24	3135	0856
185		AXA	0001		0838	50	0001	0744
186		NZA	LOOP2	SET 1	0744	40	0736	0866
187	THRU1	PCH	Y0001		1004	71	1127	0728
188		RSA	0008		0728	81	0008	0834
189		LDD	ZERO	LOOP3	0834	69	0083	0786
190	LOOP3	STD	Y0009	A	0786	24	3135	0888
191		AXA	0001		0888	50	0001	0794
192		NZA	LOOP3		0794	40	0786	0048
193		LDD	M		0048	69	1951	1054
194		RAB	8001		1054	82	8001	1110
195		RSC	0000	SET 2	1110	89	0000	0916
196	SET 2	RSA	0008	LOOP4	0916	81	0008	1022
197	LOOP4	LDD	T0001	C	1022	69	6631	0884
198		STD	Y0009	A	0884	24	3135	0938
199		AXC	0001		0938	58	0001	0844
200		SXB	0001		0844	53	0001	0900
201		NZB		THRU2	0900	42	0903	1104
202		AXA	0001		0903	50	0001	0759
203		NZA	LOOP4		0759	40	1022	0763
204		PCH	Y0001		0763	71	1127	0778
205		RSA	0008		0778	81	0008	0934
206		LDD	ZERO	LOOP5	0934	69	0083	0836
207	LOOP5	STD	Y0009	A	0836	24	3135	0988
208		AXA	0001		0988	50	0001	0894
209		NZA	LOOP5	SET 2	0894	40	0836	0916
210	THRU2	PCH	Y0001		1104	71	1127	0828
211		RSA	0008		0828	81	0008	0984
212		LDD	ZERO	LOOP6	0984	69	0083	0886
213	LOOP6	STD	Y0009	A	0886	24	3135	1038
214		AXA	0001		1038	50	0001	0944
215		NZA	LOOP6	PT25	0944	40	0886	0098
216	PT25	RAU	X0001		0098	60	1077	0881
217		FSB	X0002		0881	33	1078	1055
218		BMI	PT29	PT26	1055	46	0858	0809
219	PT26	RAU	W		0809	60	0862	0817
220		SUP	ONE		0817	11	0722	0878
221		STU	W		0878	21	0862	0065
222		STU	Q0001		0065	21	1027	0880
223		PCH	Q0001	PT27	0880	71	1027	0928
224	PT27	NZU	PT30	HALT	0928	44	0931	0000
225	PT29	RAU	M		0858	60	1951	0813
227		STU	W	PT30	0813	21	0862	0931
228	PT30	RAU	R		0931	60	0808	0863
229		SUP	ONE		0863	11	0722	0978
230		STU	R		0978	21	0808	0912
231		NZU	PT32	PT31	0912	44	0715	0966
232	PT31	LDD	N		0966	69	1952	1155
233		STD	R	PT3	1155	24	0808	0080
234	PT32	LDD	R		0715	69	0808	0962
235		RAC	8001		0962	88	8001	0008
236		LDD	ONE		0068	69	0722	0825
237		STD	T0000	C PT4	0825	24	6630	0730
238	HALT	01	9999	9999	0000	01	9999	9999
239	ONE	00	0000	0001	0722	00	0000	0001
240	FLONE	10	0000	0051	0014	10	0000	0051
241	ZERO	00	0000	0000	0083	00	0000	0000