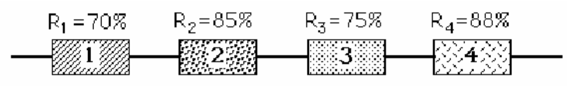
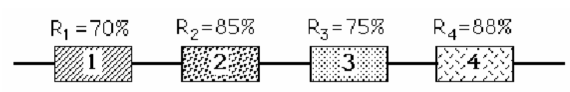
Optimal Redundancy to Maximize System Reliability

- *Dynamic Programming Model*
- *Integer Programming Model*

© D. L. Bricker
Dept of Mechanical & Industrial Engineering
The University of Iowa

One of the systems of a communication satellite consists of four unreliable components each of which are necessary for successful operation of the satellite—the probabilities that a component survives the planned lifetime of the satellite (i.e., the *reliabilities*) are shown below:





Assuming that component failures are independent,

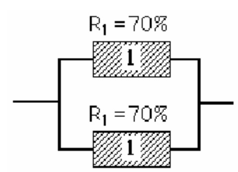
Reliability of system

= P{components 1 through 4 survive}

= P{#1 survives} × P{#2 survives} × P{#3 survives} × P{#4 survives}

= 0.70 × 0.85 × 0.75 × 0.88 = **39.27%**

This is an unacceptably low system reliability, and so redundant units of one or more components will be used in the design.



The reliability of a component may be increased by including redundant units!

Reliability of component #1

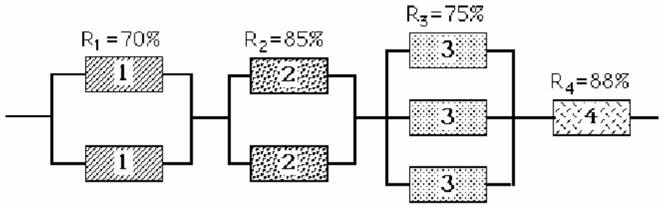
= P{at least one unit survives}

= 1 – P{both units fail}

= 1 – 0.30 × 0.30 = 91%

This assumes what is referred to as "hot standby", i.e., a standby unit may fail even before it is put into service!

By using redundant units of each component, the system reliability can be dramatically increased—for example:



$$\left\{\begin{matrix}\text{System}\\ \text{Reliability}\end{matrix}\right\} = \left[1-(0.30)^2\right]\times\left[1-(0.15)^2\right]\times\left[1-(0.25)^2\right]\times[0.88]$$

$$= 0.91\times 0.9775\times 0.984375\times 0.88 = 77.0551\%$$

The problem faced by the designer is to maximize the system reliability, subject to a restriction on the total weight of the system.

Component	1	2	3	4
Weight (kg)	1	2	1	3

Total weight must not exceed 12 kg.

(Total weight of one unit of each component is 7 kg, leaving 5 kg for redundant units.)

Reliability (%) vs. # redundant units

Component	1 unit	2 units	3 units
1	70	91	97.3
2	80	97.75	99.6625
3	75	93.75	98.4375
4	88	98.56	99.8272

We will assume that no more than three units of any component will be included!

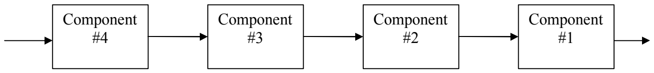
Dynamic Programming Model

Stage: n component type

Decision: x_n # of units of component n included in system

State: s_n slack weight, i.e., # kg available

We impose a sequential decision-making structure on the problem by supposing that we consider the components one at a time, deciding how many units to include based upon the available weight capacity.

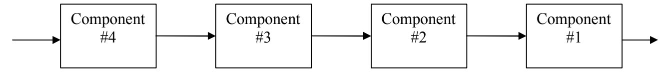


Arbitrarily we will use a "backward" order in what follows!

That is, imagine that we first consider how many units of component #4 are to be included when we begin with 12 kg of available capacity, while component #1 is the last to be considered.

Optimal Value Function

$f_n(s_n)$ = maximum reliability of the subsystem consisting of devices n, n-1, ... 1, if s_n kg of available capacity remains to be allocated.



Recursive definition of function

$$f_n(s_n) = \underset{1 \le x_n \le s_n / w_n}{\text{maximum}} \left\{ \left(1 - p_n^{x_n}\right) \times f_{n-1}\left(s_n - w_n x_n\right) \right\}$$

$$f_0(s_0) = \begin{cases} 1 & \text{if } s_0 \ge 0 \\ 0 & \text{otherwise} \end{cases}$$

APL function definition

```
   ∇ z←F N;t
     ⍝
     ⍝ Optimal redundancy to maximize reliability
     ⍝
     :if N=0
       z←((⍴s)⍴1),-BIG
     :else
     ⍝ Recursive definition of optimal value function
       z←Maximize (((⍴s)⍴1)∘.-(1-R[N])*x)×(F N-1)[TRANSITION s∘.-W[N]×x
     :endif
   ∇
```


Component #1: reliability = 70%, weight = 1 kg.

Stage 1

s \ x:	1	2	3	Maximum
1	0.7000	¯99.9999	¯99.9999	0.7000
2	0.7000	0.9100	¯99.9999	0.9100
3	0.7000	0.9100	0.9730	0.9730

etc.

Component #2: reliability = 80%, weight = 2 kg.

Stage 2

s \ x:	1	2	3	Maximum
3	0.5600	¯99.9999	¯99.9999	0.5600
4	0.7280	¯99.9999	¯99.9999	0.7280
5	0.7784	0.6720	¯99.9999	0.7784
6	0.7784	0.8736	¯99.9999	0.8736
7	0.7784	0.9341	0.6944	0.9341
8	0.7784	0.9341	0.9027	0.9341

etc.

For example, suppose that we have 6 kg of capacity remaining, i.e., $s_2 = 6$, and we choose to include 2 units of component #2. Then we obtain 97.75% reliability of subsystem #2 and arrive at stage 1 (component #1) with 6-2×2=2 kg of capacity remaining, so that we can achieve 91% reliability ($f_1(2)=0.91$) in subsystem #1. Hence the subsystem of components 1&2 will have reliability 0.9775×0.91 = 0.8736

Component #3: reliability = 75%, weight = 1 kg.

Stage 3

s \ x:	1	2	3	Maximum
4	0.4200	¯99.9999	¯99.9999	0.4200
5	0.5460	0.5250	¯99.9999	0.5460
6	0.5838	0.6825	0.5513	0.6825
7	0.6552	0.7298	0.7166	0.7298
8	0.7006	0.8190	0.7662	0.8190
9	0.7006	0.8757	0.8600	0.8757

etc.

Component #4: reliability = 88%, weight = 3 kg.

Stage 4

s \ x:	1	2	3	Maximum
7	0.3696	¯99.9999	¯99.9999	0.3696
8	0.4805	¯99.9999	¯99.9999	0.4805
9	0.6006	¯99.9999	¯99.9999	0.6006
10	0.6422	0.4140	¯99.9999	0.6422
11	0.7207	0.5381	¯99.9999	0.7207
12	0.7706	0.6727	¯99.9999	0.7706

Only the last row of this table need be computed to find the optimal reliability with 12 kg of capacity!

Summary of computations

Stage 4

Current State	Optimal Decision	Optimal Value	Next State
cap 7	1 units	0.3696	cap 4
cap 8	1 units	0.4805	cap 5
cap 9	1 units	0.6006	cap 6
cap 10	1 units	0.6422	cap 7
cap 11	1 units	0.7207	cap 8
cap 12	1 units	0.7706	cap 9

Stage 2

Current State	Optimal Decision	Optimal Value	Next State
cap 3	1 units	0.5600	cap 1
cap 4	1 units	0.7280	cap 2
cap 5	1 units	0.7784	cap 3
cap 6	2 units	0.8736	cap 2
cap 7	2 units	0.9341	cap 3
cap 8	2 units	0.9341	cap 4

Stage 3

Current State	Optimal Decision	Optimal Value	Next State
cap 4	1 units	0.4200	cap 3
cap 5	1 units	0.5460	cap 4
cap 6	2 units	0.6825	cap 4
cap 7	2 units	0.7298	cap 5
cap 8	2 units	0.8190	cap 6
cap 9	2 units	0.8757	cap 7

Stage 1

Current State	Optimal Decision	Optimal Value	Next State
cap 1	1 units	0.7000	cap 0
cap 2	2 units	0.9100	cap 0
cap 3	3 units	0.9730	cap 0
cap 4	3 units	0.9730	cap 1
cap 5	3 units	0.9730	cap 2
cap 6	3 units	0.9730	cap 3

The maximum reliability, then, given a 12 kg weight restriction, is $f_4(12) = 77.06\,\%$

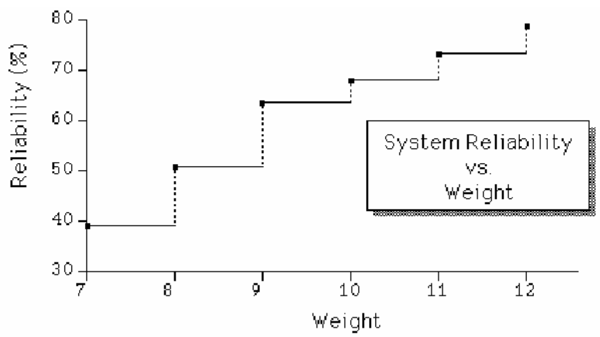
By a "forward pass" through the tables, we can determine the optimal design:

stage	state	decision
4	cap 12	1 units
3	cap 9	2 units
2	cap 7	2 units
1	cap 3	3 units
0	cap 0	

That is, the optimal design includes 1 of component #4, 2 each of components #2 & #3, and 3 of component #1.

- What reduction in reliability would occur if the weight restriction were 11 kg rather than 12?

- What is the optimal design with a weight restriction of 11 kg?



Integer Programming Model

Define *binary* decision variables:

X_{in} = 1 if *n* units of component *i* are included in the system

X_{in} = 0 otherwise

Notation:

Component i	R_{i1}	R_{i2}	R_{i3}
1	0.70	0.91	0.973
2	0.80	0.9775	0.996625
3	0.75	0.9375	0.984375
4	0.88	0.9856	0.998272

Objective:

In order to linearize the objective, we will instead maximize the ***logarithm of the reliability***:

$$Maximize \sum_{i=1}^{4}\sum_{n=1}^{3}(\ln R_{in})X_{in}$$

subject to

$$\sum_{i=1}^{4}\sum_{n=1}^{3}(W_i n)X_{in} \le W_{max}$$

$$\sum_{n=1}^{3} X_{in} = 1 \qquad \forall i = 1,2,3,4$$

$$X_{in} \in \{0,1\} \qquad \forall i \& n$$

Component i	*ln* R_{i1}	*ln* R_{i2}	*ln* R_{i3}
1	-0.35667	-0.094311	-0.02737
2	-0.22314	-0.040822	-0.008032
3	-0.28768	-0.064539	-0.01575
4	-0.12783	-0.014505	-0.001729

LINGO model:

```
SETS:
  COMPONENT / A B C D/:
    WEIGHT;
  UNITS / 1..3/;
  LOG(COMPONENT,UNITS): LNR, X;
ENDSETS

DATA:
  WEIGHT = 1 2 1 3;
  WMAX = 12;
  LNR = -0.35667 -0.094311 -0.027371
        -0.22314 -0.040822 -0.0080322
        -0.28768 -0.064539 -0.015748
        -0.12783 -0.014505 -0.0017295; ! LNR is log of reliability;
ENDDATA

MAX = @SUM( COMPONENT(I): @SUM(UNITS(N):LNR(I,N)*X(I,N))) ;

@SUM( COMPONENT(I): @SUM(UNITS(N): WEIGHT(I)*N*X(I,N)))<= WMAX;

@FOR (COMPONENT(I):
  @SUM (UNITS(N): X(I,N))=1;       );
@FOR (COMPONENT(I):
  @FOR (UNITS(N):  @BIN (X(I,N)) ) );
```


LINDO model:

```
MAX   - .35667 X( A, 1) - .094311 X( A, 2) - .027371 X( A, 3)
      - .22314 X( B, 1) - .040822 X( B, 2) - .0080322 X( B, 3)
      - .28768 X( C, 1) - .064539 X( C, 2) - .015748 X( C, 3)
      - .12783 X( D, 1) - .014505 X( D, 2) - .0017295 X( D, 3)
 SUBJECT TO
 2]  X( A, 1) + 2 X( A, 2) + 3 X( A, 3) + 2 X( B, 1) + 4 X( B, 2)
      + 6 X( B, 3) + X( C, 1) + 2 X( C, 2) + 3 X( C, 3) + 3 X( D, 1)
      + 6 X( D, 2) + 9 X( D, 3) <=   12
 3]  X( A, 1) + X( A, 2) + X( A, 3) =    1
 4]  X( B, 1) + X( B, 2) + X( B, 3) =    1
 5]  X( C, 1) + X( C, 2) + X( C, 3) =    1
 6]  X( D, 1) + X( D, 2) + X( D, 3) =    1
 END
 INTE    12
```

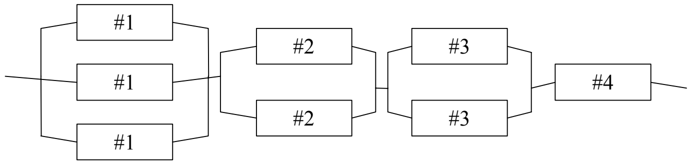
Optimal Solution:

```
Objective value:    - 0.2605620

Variable            Value         Reduced Cost
X( A, 3)          1.000000           0.2737100E-01
X( B, 2)          1.000000           0.4082200E-01
X( C, 2)          1.000000           0.6453900E-01
X( D, 1)          1.000000           0.1278300
```

Note that $\exp\{-0.2605620) = 0.77062$

which is in agreement with the dynamic programming solution.



Optimal Design