



# Integrated Medical Model (IMM) Optimization Version 4.0 Functional Improvements

2016 Human Research Program Investigators' Workshop

Monday, February 8, 2016

Presenter:

John Arellano, Ph.D. – IMM Lead Developer

M. Young, L. Boley, Y. Garcia, L. Saile, M. Walton, E. Kerstman, D. Reyes,  
D. A. Goodenow, and J. G. Myers



- **Output**
  - Predicted Medical Events (What we think will happen)
  - Required Resources (What we have available for treatment)

# Now What???



- **Is it possible to pack all the resources used for treatment?**
- **If not, how many resources can be utilized?**
- **What are the most useful resources?**
- **Maximize Crew Health Index (CHI)**
- **Minimize probability of Evacuation (EVAC)**
- **Minimize probability of Loss of Crew Life (LOCL)**

# The 0/1 Knapsack Problem



$$\begin{aligned} \max \quad & \sum_{i=1}^n b_i x_i \\ \text{s. t.} \quad & \sum_{i=1}^n w_i x_i \leq C \end{aligned}$$

**where**

$b_i$  is the benefit of having item  $i$  in the knapsack

$w_i$  is the weight of item  $i$

$C$  is the weight limit on the knapsack

$x_i$  is a binary variable for item  $i$  that is zero if item  $i$  is not in the knapsack and one if it is in the knapsack.

$$\begin{aligned} \max \quad & \sum_{i=1}^n b_i x_i \\ \text{s. t.} \quad & \sum_{i=1}^n v_i x_i \leq V \\ & \sum_{i=1}^n m_i x_i \leq M \end{aligned}$$

**where**

$b_i$  is the benefit of having item  $i$

$v_i$  is the volume of item  $i$

$m_i$  is the mass of item  $i$

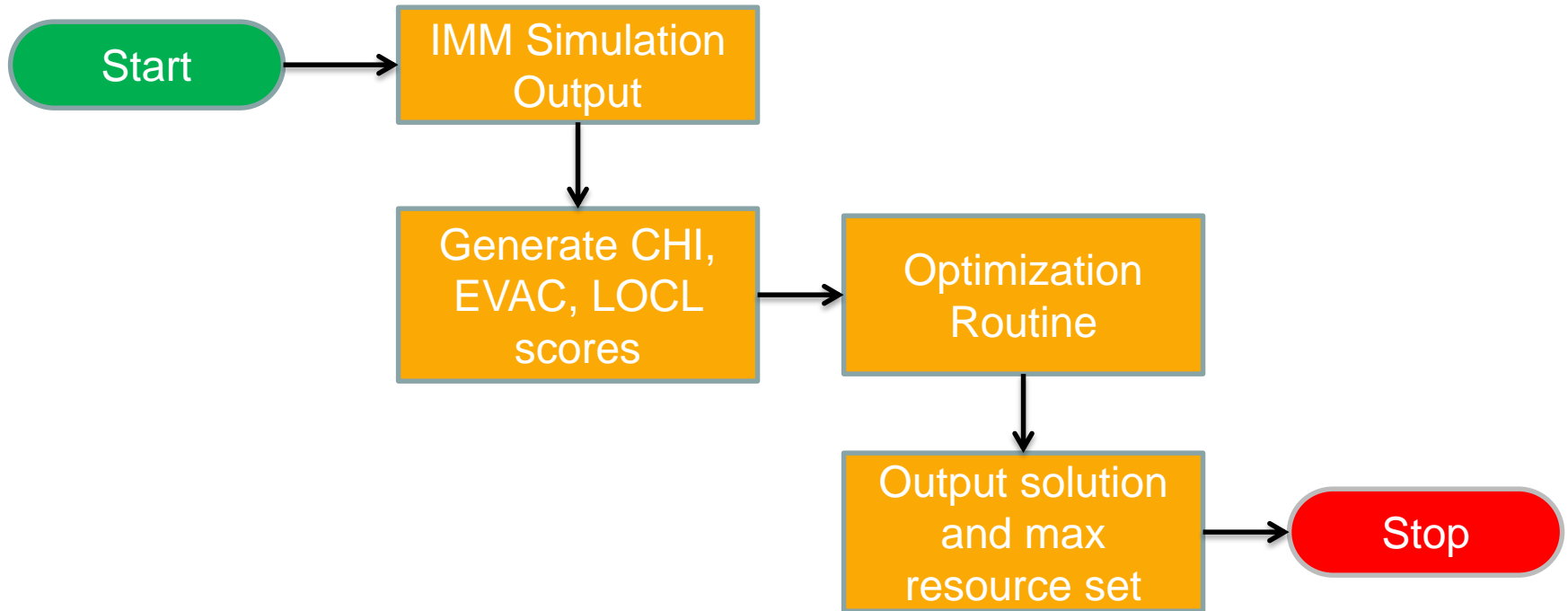
$V$  is the volume limit

$M$  is the mass limit

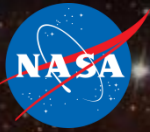
$x_i \in \{0,1\}$  for all items  $i$

An item is a discretized unit for each resource

# IMM 4.0 Optimization



# Functional Improvements from 3.0 to 4.0



- **Granularity of solutions**
- **Improved score generation**
- **Ability to generate solutions based on different combinations of mass, volume, and optimization priority automatically.**



# Granularity of Solutions



CONDITION	RESOURCES	BEST	WORST	
BACK SPRAIN/STRAIN	TYLENOL (ACETAMINOPHEN) 325 MG	8	45	PER DAY
BACK SPRAIN/STRAIN	VICODIN HP (HYDROCODONE/ ACETAMINOPHEN HP) 10 MG/660 MG	0	30	PER DAY
BACK SPRAIN/STRAIN	MOTRIN (IBUPROFEN) 400MG	4	15	PER DAY

- **Version 3.0**

- All or nothing, an item is considered an entire treatment
- Enough space for the entire treatment is needed to add the treatment to the set of resources

- **Version 4.0**

- An item is considered the smallest discretized unit for each particular resource
- Can add individual items to the set of resources without requiring an entire treatment

- **Advantage**

- Represents the real world system more accurately





- **Score generation is the most computationally expensive**
- **Version 3.0**
  - An item is a treatment
  - Score generation took about 5 hours for 100,000 trials
  - After some code tinkering, 15 minutes for 100,000 trials
- **Version 4.0**
  - An item is a discretized unit of a resource
  - Score generation took about 5 hours for 100,000 trials
  - After some code tinkering (Parallel Computing Toolbox), 10 minutes for 100,000 trials
- **Advantage**
  - Ability to generate solutions faster



- **The optimization routine has been updated to use arrays of Mass, Volume and Optimization Priority**
- **Version 3.0**
  - Mass = 13 kg
  - Volume = 8200 cc
  - Maximize CHI
- **Version 4.0**
  - Mass = [10, 15, 20] kg
  - Volume = [7500, 8200, 9000] cc
  - Maximize CHI, Minimize EVAC and LOCL
- **Advantage**
  - Ability to generate solutions based on different input parameters automatically



- **Example DRM Profile**

- 14 day mission
- 4 Crew members
  - 1 Female
  - 1 Crown
  - No EVAs
  - 2 CAC scores greater than zero
  - 1 abdominal surgery
- Mass = [5, 9, 14] kg, Volume = [18288, Infinite] cc, maximize CHI

- **Initial IMM Simulation**

- **Optimization is performed for different combinations**

- **IMM Simulations are performed to compare the CHI of the various kits to the ISS baseline**

# Conclusion



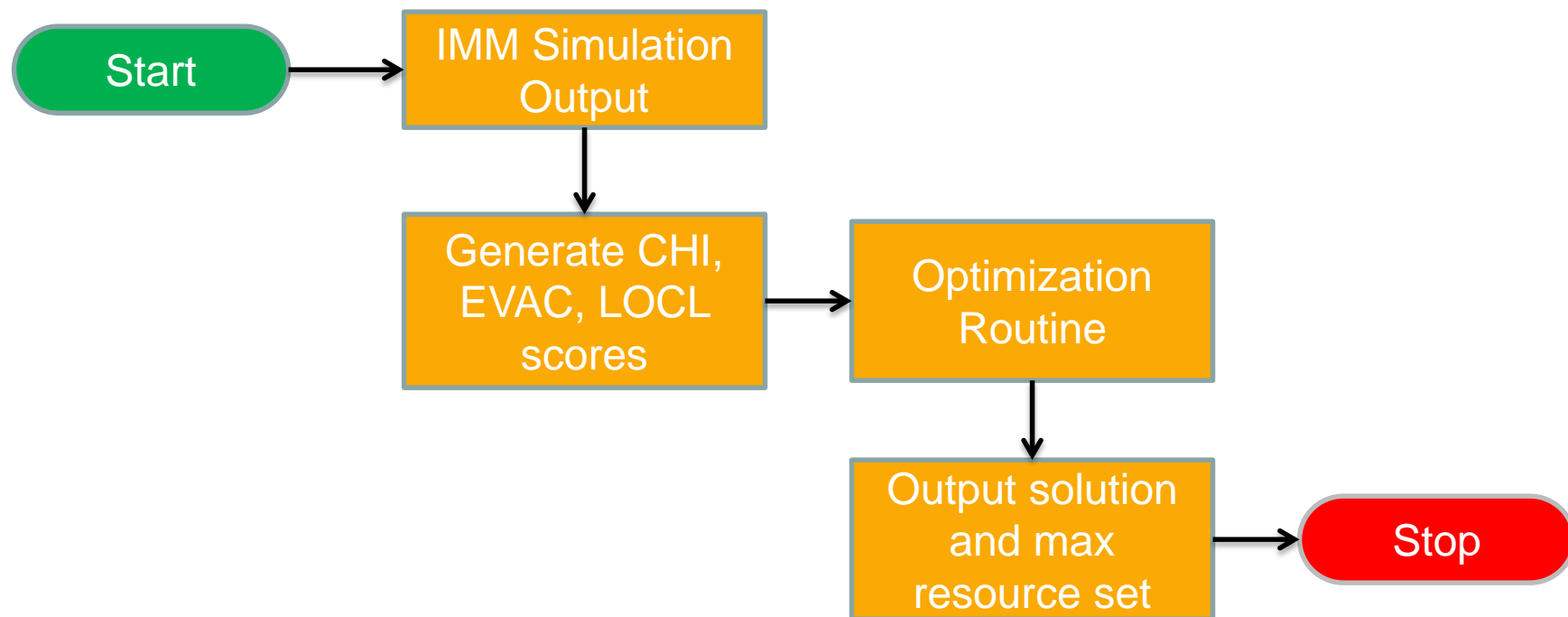
- **The greatest performance improvement stemmed from the score generation for CHI, EVAC, and LOCL**
- **By taking advantage of the structure of the output, and MATLAB's Parallel Computing Toolbox, the optimization routine now runs more efficiently**





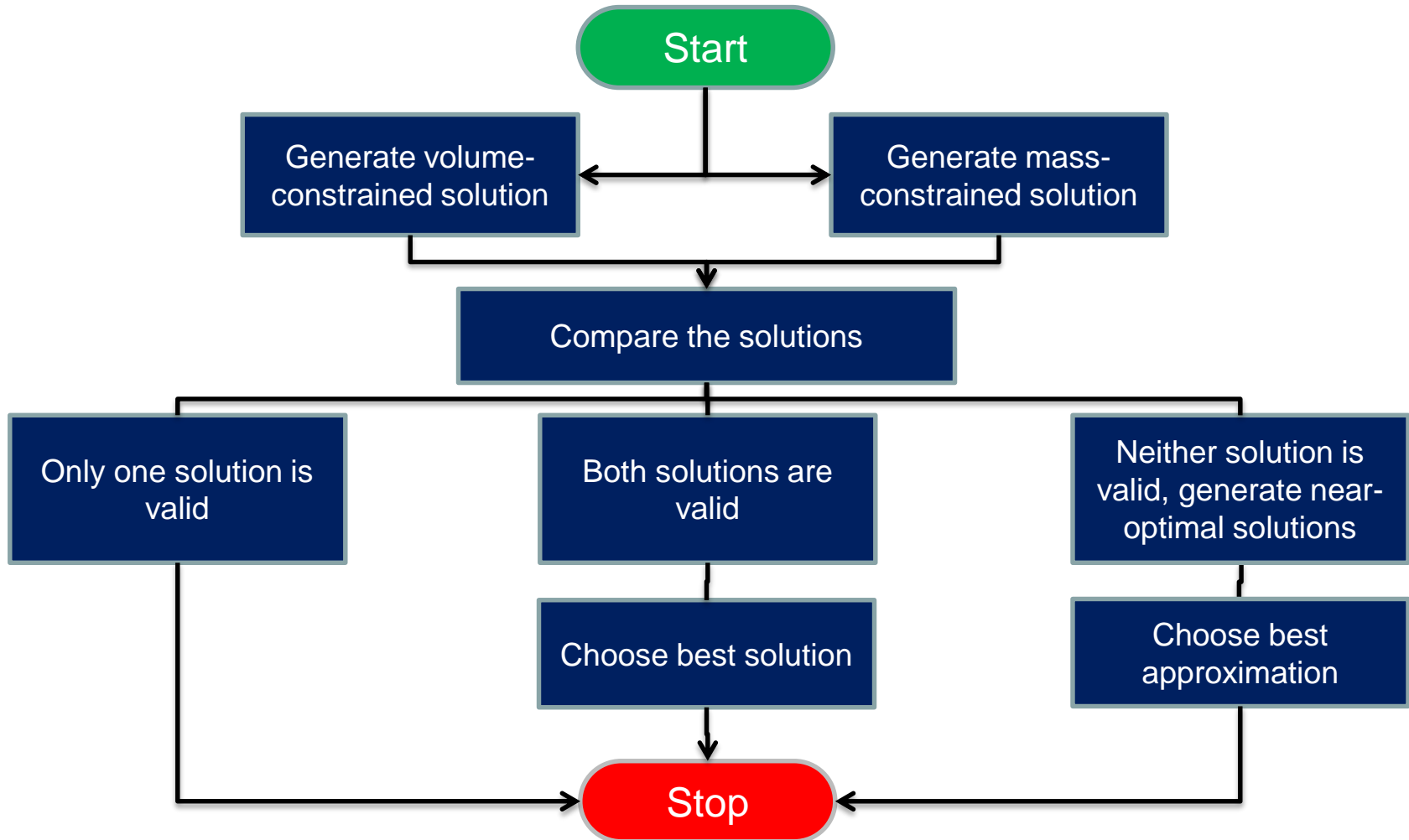
# Extra Slides

# IMM 4.0 Optimization

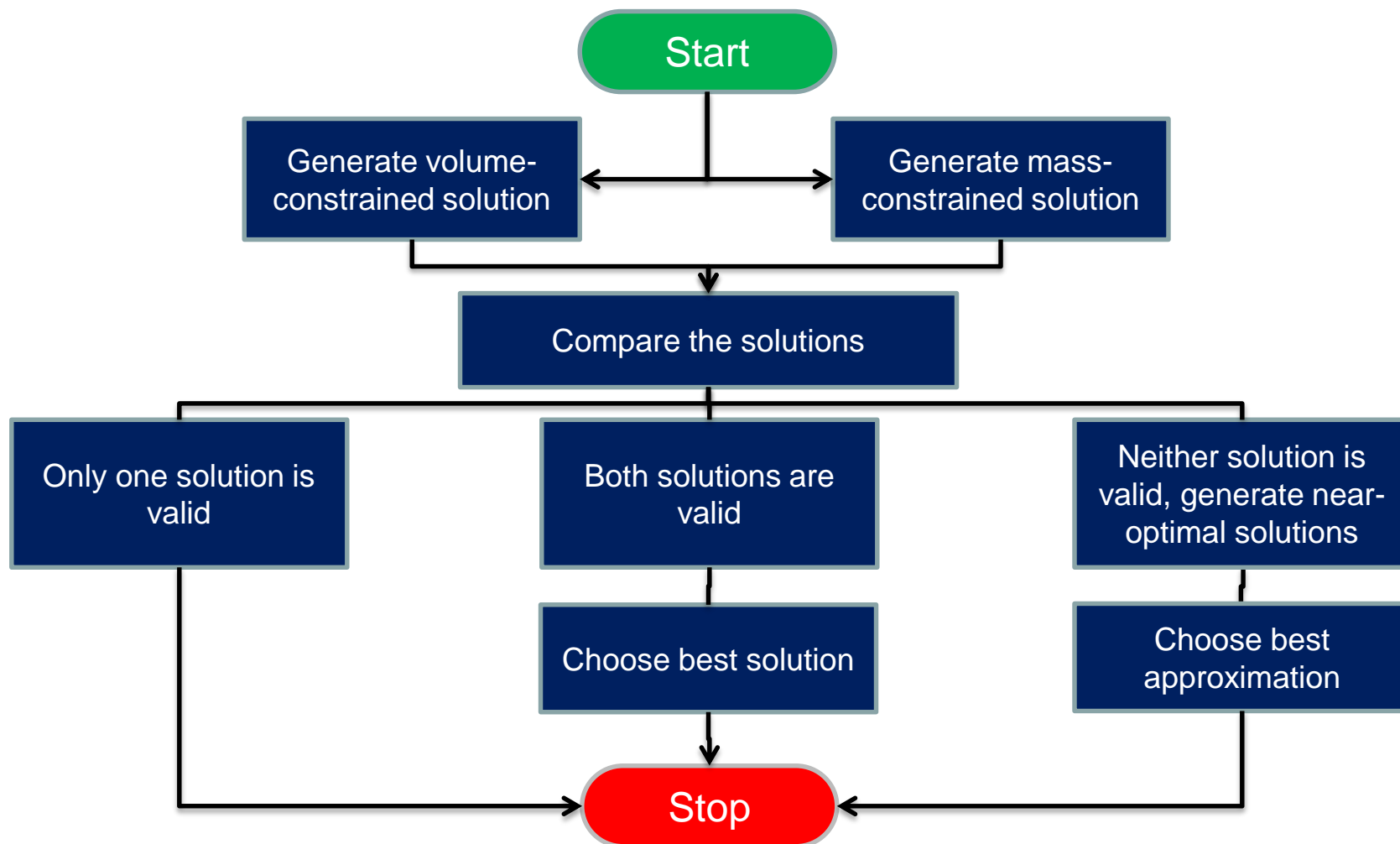




# The Optimization Routine



# The Optimization Routine





- For each volume constraint
  - For each mass constraint
    - For each OPTIMIZATION PRIORITY
      - Generate volume-constrained solution
      - Generate mass-constrained solution
      - If both solutions are valid
        - » choose the best one
      - Elseif only the volume-constrained solution is valid
        - » Choose the volume-constrained solution
      - Elseif only the mass-constrained solution is valid
        - » Choose the volume-constrained solution
      - Elseif both are invalid
        - » Generate near-optimal solutions and choose the best one
      - Endif
    - EndFor
  - EndFor
- Endfor

- **Group medical events over all the trials and get the difference between the means**

```
tx_idx = Event.TXvalid==1;
TXEvent1 = dataset(Event(tx_idx,:).Medical_Condition,Event(tx_idx,:).scenario,...
    Event(tx_idx,:).TXQTL, Event(tx_idx,:).TXEVAC,Event(tx_idx,:).TXLOCL);
TXEvent1.Properties.VarNames = {'MedicalCondition', 'scenario',...
    'TXQTL', 'TXEVAC', 'TXLOCL'};
TXstat = grpstats(TXEvent1,{'MedicalCondition', 'scenario'},...
    'mean', 'DataVars',{'TXQTL','TXEVAC','TXLOCL'});

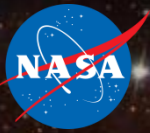
ux_idx = Event.UXvalid==1;
UXEvent1 = dataset(Event(ux_idx,:).Medical_Condition,Event(ux_idx,:).scenario,...
    Event(ux_idx,:).UXQTL, Event(ux_idx,:).UXEVAC,Event(ux_idx,:).UXLOCL);
UXEvent1.Properties.VarNames = {'MedicalCondition', 'scenario',...
    'UXQTL', 'UXEVAC', 'UXLOCL'};
UXstat = grpstats(UXEvent1,{'MedicalCondition', 'scenario'},...
    'mean', 'DataVars',{'UXQTL','UXEVAC','UXLOCL'});

Eventstat = join(UXstat, TXstat);
Eventstat.delta_mean_QTL = Eventstat.mean_UXQTL - Eventstat.mean_TXQTL;
Eventstat.delta_mean_EVAC = Eventstat.mean_UXEVAC - Eventstat.mean_TXEVAC;
Eventstat.delta_mean_LOCL = Eventstat.mean_UXLOCL - Eventstat.mean_TXLOCL;
```

- Compute the resource proportional scores

$$\text{resourceProportion}(i) = \frac{1}{\text{ceil}\left(\text{resourceCount}(i) * \left(\frac{\text{total}(i)}{\text{min fraction}(i)}\right)\right)}$$

Condition	Number of Unique Resources	Resources	Total	Min Fraction
Condition A	2	Resource 1	1	1
Condition A	2	Resource 2	3	1
Condition B	3	Resource 3	2	1
Condition B	3	Resource 4	1	1
Condition B	3	Resource 5	4	1



- **Compute the resource scores for each item**

$$\text{resourceScoreQTL}(i) = \text{deltaMeanQTL}(i) * \text{resourceProportion}(i)$$

$$\text{resourceScoreEVAC}(i) = \text{deltaMeanEVAC}(i) * \text{resourceProportion}(i)$$

$$\text{resourceScoreLOCL}(i) = \text{deltaMeanLOCL}(i) * \text{resourceProportion}(i)$$

# Generate Scores



```
BCWCResources.resourceProportion = zeros(length(BCWCResources.resourceID),1);
BCWCResources.resourceProportion =...
    1./(BCWCResources.resourceCount.*ceil(BCWCResources.total./BCWCResources.min_fraction));

BCWCResources.delta_mean_QTL = zeros(length(BCWCResources),1);
BCWCResources.delta_mean_EVAC = zeros(length(BCWCResources),1);
BCWCResources.delta_mean_LOCL = zeros(length(BCWCResources),1);

for i=1:length(Eventstat)
    temp_idx = BCWCResources.scenario == Eventstat.scenario(i) & ...
        strcmp(Eventstat.Medical_Condition(i),BCWCResources.Medical_Condition);
    BCWCResources(temp_idx,:).delta_mean_QTL = Eventstat.delta_mean_QTL(i).*ones(sum(temp_idx),1);
    BCWCResources(temp_idx,:).delta_mean_EVAC = Eventstat.delta_mean_EVAC(i).*ones(sum(temp_idx),1);
    BCWCResources(temp_idx,:).delta_mean_LOCL = Eventstat.delta_mean_LOCL(i).*ones(sum(temp_idx),1);
end

BCWCResources.resourceScoreQTL = BCWCResources.delta_mean_QTL .* BCWCResources.resourceProportion;
BCWCResources.resourceScoreEVAC = BCWCResources.delta_mean_EVAC .* BCWCResources.resourceProportion;
BCWCResources.resourceScoreLOCL = BCWCResources.delta_mean_LOCL .* BCWCResources.resourceProportion;
```





- **Initialize the maximum resource set with CHI, EVAC, and LOCL scores**
- **For each trial t**
  - For each medical event C
    - Create temporary trial resource set
    - For each resource that treats medical event C
      - Create new scores for consumable resources
      - Create/Update scores for non-consumable resources
    - Endfor
  - Endfor
  - For each resource in the trial resource set
    - Update the resource in the maximum resource set
  - Endfor
- **Endfor**



# Generating the max resource set scores

## Version 1

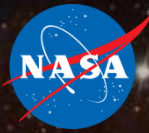
- Create an empty max resource set
- For each trial t
  - Create a trial resource set
  - Get the trial information from Event and summarize it
  - For each medical event
    - For each resource used to treat the medical event
      - Update the scores in the trial resource set
  - For each resource in the trial resource set
    - Update the corresponding scores in the max resource set

## Version 2

- Create an empty max resource set
- Separate Event by trial
- Create main TRIAL RESOURCE SET
- Parfor each trial t
  - Retrieve Event information by trial
  - Create trial resource set
  - For each medical event
    - For each resource used to treat the medical event
      - Update the scores in trial resource set
  - Update TRIAL RESOURCE SET with trial resource set
- Update max resource set using TRIAL RESOURCE SET

The sum of the differences for the CHI, EVAC, and LOCL scores are on the order of  $1e-06$ ,  $1e-10$ ,  $1e-12$  respectively.

# Requirements Verification



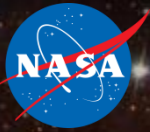
Requirement Number	Requirement Title	Requirement Description	Interpretation
R1	Maximize Crew Health Index (CHI)	The IMM medical kit optimization routine shall generate a medical kit subject to user-specified mass and/or volume constraints that maximizes the CHI for a specified mission and crew profile.	The optimization routine shall provide an optimal or near-optimal set of resources based on the CHI scores
R2	Minimize probability of Evacuation (EVAC)	The IMM medical kit optimization routine shall generate a medical kit subject to user-specified mass and/or volume constraints that minimizes the probability of consideration of EVAC for a specified mission and crew profile.	The optimization routine shall provide an optimal or near-optimal set of resources based on the EVAC scores
R3	Minimize probability of Loss of Crew Life (LOCL)	The IMM medical kit optimization routine shall generate a medical kit subject to user-specified mass and/or volume constraints that minimizes the probability of LOCL for a specified mission and crew profile.	The optimization routine shall provide an optimal or near-optimal set of resources based on the LOCL scores
R4	IMM Interface	The IMM medical kit optimization routine shall interface with the IMM output datasets in such a way as to use them in objectively generating optimal medical kits.	The optimization routine shall objectively score the set of available resources based on the optimization priority, CHI, EVAC, or LOCL.

# Generating a Solution



- **To generate a solution, a dynamic programming algorithm for a single constraint 0/1 Knapsack Problem is implemented**
- **Build an optimal solution from subproblems using a 2D array for storage.**

# There Are Two Constraints???

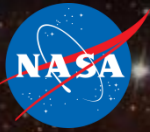


- **Why not use dynamic programming for multi-constrained problems? For example,  $M(i, m, v)$ , a 3D matrix.**
- **Answer: Not enough bang for the buck.**
- **Too much computational time?**
- **Would need to run 1 optimization for each incremental volume/mass. For example, 18288 cc/13.6kg.**
  - 182,880 mass-constrained problems (0.5 secs each)~25 hours
  - 1,360 volume-constrained problems (7 secs each)~3 hours

# Dynamic Programming for 0/1 the Knapsack Problem



- A matrix,  $M$ , is used to hold all computed values during the optimization technique
- Entry  $M(i, w)$  is generated using the following step:
  - $M(i, w) = M(i - 1, w)$  if  $w_i > w$   $M(i, w) = \max(M(i - 1, w), M(i - 1, w - w_i) +$



- **Want to pack most beneficial resources, “most bang for your buck”**
- **Therefore, try to provide the most beneficial resources that fit.**
- **What does it mean to be beneficial?**
- **Maximize CHI, Minimize pEVAC/pLOCL**