Literature Review of Weapon Target Assignment problem

Guanda Li joint work with Liang Chen and Yu-hong Dai

Institute of Computational Mathematics and Scientific/Engineering Computing,
Academy of Mathematics and Systems Science,
Chinese Academy of Sciences

December 20, 2022

Contents

Introduction

Background

背景介绍

- **Weapon Target Assignment problem** 是一个军事领域中的问题
- 其基本问题是考虑使用 m 个武器攻击 n 个目标，以最小化所有目标的加权存活概率.（或等价地，对目标造成的加权伤害最大化）
- **该问题被证明是 NP-C 的**，并且通常被表述为非线性模型。
- To best of my knowledge, 2015 年前，在 80 个武器和 80 个目标的问题计算需要 16.2 个小时。

history

- 提出 WTA 问题形式 Manne (1958)
- 通过对问题进行线性近似、求解问题的简化形式与求解小规模问题
- 首次提出动态 WTA 问题 (1985)
- 改进静态与动态 WTA 问题的模型，采用启发式算法进行求解
- 对问题进行等价的线性转换并精确求解

basic formulation

- $I = \{1, \cdots, m\}$, 武器集合.
- $J = \{1, \cdots, n\}$, 目标集合.
- $p_{ij} \in [0, 1]$, i 击落 j 的概率.
- V_j, 目标 j 的权重.
- x_{ij}, 武器 i 是否攻击 j.

$$\begin{aligned} \max \quad & \sum_{j=1}^{n} (1 - V_j) \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right) \qquad \text{(S0)} \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le 1 \quad \forall\, i \in I, \\ & x_{ij} \in \{0, 1\} \quad \forall\, j \in J,\ i \in I. \end{aligned}$$

basic formulation

- $I = \{1, \cdots, m\}$, 武器集合.
- $J = \{1, \cdots, n\}$, 目标集合.
- $p_{ij} \in [0, 1]$, i 击落 j 的概率.
- V_j, 目标 j 的权重.
- x_{ij}, 武器 i 是否攻击 j.

$$\min \quad \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \tag{2}$$

$$x_{ij} \in \{0, 1\} \quad \forall\, j \in J,\ i \in I. \tag{3}$$

Formulations and algorithms

whether the problem is static

- WTA 问题主要分为了静态 WTA 问题（SWTA 问题）与动态 WTA 问题（DWTA 问题）
- 静态 WTA 问题中不考虑不同的时间阶段，仅考虑针对一次目标来袭所做的拦截。
- 动态 WTA 问题则考虑多阶段的问题，在不同的阶段呢可能有不同的目标出现，武器的分配也同样需要照顾多个不同的时间阶段。
- 近年来，也有一些新的问题形式出现，比如加入更多的作战要素或者使用多目标规划。

Static Weapon Target Assignment problem

Notation

- I, Weapon set.
- J, Target set.
- n, 目标的个数。
- m, 武器种类的个数。如果限制每种武器只有一个，则该变量指示武器的个数。
- w_i, 第 i 类武器的个数
- t, 时间阶段（仅在 DWTA 问题中有效）
- p_{ij}, 武器 i 摧毁目标 j 的概率。
- q_{ij}, 武器 i 未能摧毁目标 j 的概率。
- x_{ij}, 分配给目标 j 的种类为 i 的武器个数，如果限制每种武器仅有一个，则该变量为 0-1 变量。
- V_j, 目标 j 的摧毁价值。
- c_{ij}, 将武器 i 分配给目标 j 的花费
- s_j, 能够分配给目标 j 的最大武器数目

static WTA model 1

- 相比于基础模型，允许每种武器有 w_i 个。

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right) \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le w_i \quad \forall\, i \in I, \\
& x_{ij} \in \mathbb{Z}^{+} \quad \forall\, j \in J,\ i \in I.
\end{aligned} \tag{S1.1}
$$

static WTA model 1

- 如果将 $1-p_{ij}$ 记为 q_{ij}，可以转化为看起来更简单的形式。

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} q_j^{x_{ij}} \right) \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le w_i \quad \forall\, i \in I, \\ & x_{ij} \in \mathbb{Z}^+ \quad \forall\, j \in J,\ i \in I. \end{aligned} \tag{S1.2}$$

static WTA model 2

- 为了让计算更简便，假设任何武器攻击同一个目标的击毁概率都相同。

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j q_j^{\sum_{i=1}^{m} x_{ij}} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le w_i \quad \forall\, i \in I, \\
& x_{ij} \in \mathbb{Z}^{+} \quad \forall\, j \in J,\; i \in I.
\end{aligned} \tag{S2}
$$

static WTA model 3

- 将 S1 中的目标函数取对数，即可得到如下模型

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j \exp\left(\sum_{i=1}^{m} x_{ij} \ln(1-p_{ij})\right) && \text{(S3.1)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le w_i \quad \forall\, i \in I, \\
& x_{ij} \in \mathbb{Z}^+ \quad \forall\, j \in J,\ i \in I.
\end{aligned}
$$

static WTA model 3

- 将上述模型进行变量替换即可得到如下的模型

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j e^{y_j} && \text{(S3.2)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq w_i \quad \forall\, i \in I, \\
& \sum_{i=1}^{m} \ln(1 - p_{ij}) x_{ij} = y_j \quad \forall\, j \in J \\
& x_{ij} \in \mathbb{Z}^{+} \quad \forall\, j \in J,\ i \in I.
\end{aligned}
$$

- 该模型从数学上与模型 S1 是等价的，但是新形式更方便求解器的求解。
- Kline et al.(2017b) 指出，使用商用求解器 BARON 求解问题，该形式能够让正确率提高 21%

static WTA model 4

- 该模型与一开始介绍的基本模型相同，即限制每种武器的类型只有一个。如果某个类型的武器有不止一个，可以将它们视为不同的。
- 由于 $(1-p_{ij}x_{ij})=(1-p_{ij})^{x_{ij}}$，$x\in\{0,1\}$，可以将问题转化为如下形式。

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} (1-p_{ij})^{x_{ij}} \right) && \text{(S4)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \le 1 \quad \forall\, i \in I, \\
& x_{ij} \in \{0,1\} \quad \forall\, j \in J,\ i \in I.
\end{aligned}
$$

- 该转化会导致问题的松弛从一个凸问题变成一个非凸问题，会导致一些凸的求解方法无法使用。

static WTA model 5

- 将问题转化为背包模型

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n}\sum_{x=1}^{m} c_{ij}x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \\
& \sum_{i=1}^{m} x_{ij} \leq 1 \quad \forall\, j \in J \\
& x_{ij} \in \{0,1\} \quad \forall\, j \in J,\ i \in I.
\end{aligned} \tag{S5}
$$

- 该方法去掉了多个武器对同一个武器同时打击时的耦合，将困难的目标函数转化为了线性函数
- 虽然该方法能够比较快的求解，但是却不如之前的模型更贴近现实情况。

static WTA model 6

- This is attacking model.
- there are K assets we have to protect, and n targets try to destroy these assets.
- The probablitiy with target j will destroy k is γ_{jk}.

$$\begin{aligned} \min \quad & \sum_{k=1} K a_k \prod_{j=1}^{n_k} \left[\gamma_{ij} \prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right] \qquad \text{(S6)} \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \\ & \sum_{k=1}^{K} n_k = n \\ & x_{ij} \in \{0,1\} \quad \forall\, j \in J,\ i \in I. \end{aligned}$$

Dynamic Weapon Target Assignment problem

dynamic WTA model 1 : shoot-look-shoot

- 该模型假设可以在一次打击结束后对现状进行观察，对尚未被击毁的目标重新安排武器
- 在该模型中，不会有新的目标出现
- 将会对给定数量的目标进行反复攻击，直到全部目标都被摧毁或迭代次数到达上限

dynamic WTA model 2 : multi-stages

- 该模型假设目标出现的过程会出现多个阶段
- 仅在第一阶段出现的目标是给定的，之后阶段出现的新目标以概率分布的形式给出
- 与 look-shoot-look 模型最大的不同即为 look-shoot-look 不会出现新的目标
- 由于模型的复杂度较高，目前还没有比较好的精确解法，主要采用启发式解法

Recent Developments

Sensor WTA problems

- 背景：在现代化战争中，需要引入传感器对要打击的目标进行识别，类似地还需要引入指挥所等要素
- 下面是一个较为简单的模型：S 表示传感器的集合，o 表示传感器的数目

$$\begin{aligned}
\min \quad & \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} \prod_{s=1}^{o} (1 - p_{isj})^{x_{isj}} \right) && \text{(Sensor)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} \sum_{s=1}^{o} x_{isj} \leq 1 \quad \forall\, i \in I, \\
& \sum_{j=1}^{n} \sum_{i=1}^{m} x_{isj} \leq 1 \quad \forall\, s \in S, \\
& x_{isj} \in \{0, 1\} \quad \forall\, j \in J,\ i \in I,\ s \in S.
\end{aligned}$$

- 两个约束分别是每个武器与每个传感器仅能分配给一个目标。

Multi-objective programs

- 在真实的作战环境中，不可能仅考虑对目标的杀伤概率
- 比较常见引入的目标包括最小化花费、最小化作战时间
- 最小化花费：对目标 j 使用武器 i 需要一个固定的花费
- 最小化作战时间：使用武器攻击目标需要耗费时间，为了达到作战目的也需要最小化作战时间
- 多个目标的结合方式包括将多个目标组合在一起，或者同时优化三个不同的目标，希望找到所有非占优解。在一些论文中，还提出了算法与指挥官结合的方法，即在算法执行过程中由指挥官选择不同目标的重要性。

Outer Approximation

basic formulation

- $I = \{1, \cdots, m\}$, 武器集合.
- $J = \{1, \cdots, n\}$, 目标集合.
- $p_{ij} \in [0, 1]$, i 击落 j 的概率.
- a_j, 目标 j 的权重.
- x_{ij}, 武器 i 是否攻击 j.

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} a_j \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right) \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \\ & x_{ij} \in \{0, 1\} \quad \forall\, j \in J,\ i \in I. \end{aligned} \tag{S0}$$

formulation

$$\prod_{i=1}^{m}(1-p_{ij}x_{ij}) \iff \prod_{i=1}^{m}(1-p_{ij})^{x_{ij}}, x \in \{0,1\}$$

- 当两个问题限定 x 取值为 $0,1$ 时，取值相等

目标函数为多个凸函数的加权和

采用后一种形式：

$$f_j(x) = \prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \quad \forall\, j \in J$$

考虑该问题的 Hessian 矩阵 H：

$$\frac{\partial f_j(x)}{\partial x_{aj} \partial x_{bj}} = \ln(1 - p_{aj}) \ln(1 - p_{bj}) f_j(x)$$

$$H = f(x) \begin{bmatrix} \ln(1-p_{1j})\ln(1-p_{1j}) & \ln(1-p_{1j})\ln(1-p_{2j}) & \cdots & \ln(1-p_{1j})\ln(1-p_{mj}) \\ \ln(1-p_{2j})\ln(1-p_{1j}) & \ln(1-p_{2j})\ln(1-p_{2j}) & \cdots & \ln(1-p_{2j})\ln(1-p_{mj}) \\ \vdots & \vdots & \ddots & \vdots \\ \ln(1-p_{mj})\ln(1-p_{1j}) & \ln(1-p_{mj})\ln(1-p_{2j}) & \cdots & \ln(1-p_{mj})\ln(1-p_{mj}) \end{bmatrix}$$

目标函数为多个凸函数的加权和

若记

$$l = \begin{bmatrix} \ln(1-p_{1j}) & \ln(1-p_{2j}) & \cdots & \ln(1-p_{mj}) \end{bmatrix}$$

可得到 Hassien 矩阵的形式为:

$$H = f(x) l \cdot l^T$$

可以看出该矩阵是个秩一矩阵，即各列之间是线性相关的，从而其行列式为 0，从而得到原函数是凸的。

transformed model

原模型的目标函数为:

$$\sum_{j=1}^{n} a_j \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right)$$

希望通过引入辅助变量的形式，将非线性项从目标函数转化到约束中。
若记 η_j 为代表 $\prod_{i=1}^{m}(1 - p_{ij})^{x_{ij}}$ 引入的变量，原问题可转化为:

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} a_j \eta_j \\ \text{s.t.} \quad & \eta_j \geq \prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}}, \quad \forall\, j \in J \\ & \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \\ & x_{ij} \in \{0, 1\} \quad \forall\, j \in J \quad i \in I. \end{aligned} \tag{S0'}$$

Basic idea of outer approximation

若 $f(x)$ 为凸函数，则对于可行域中的任意一个点 x^*，有

$$f(x) \geq f(x^*) + \nabla f(x^*)(x - x^*)$$

因此若原问题的约束为 $\eta \geq f(x)$，则此时

$$\eta \geq f(x) \geq f(x^*) + \nabla f(x^*)(x - x^*)$$

一定成立。

- 即对于任意一个给定的点，都可以引入一个线性约束，保证可行域满足该约束。
- 外逼近方法即希望利用整数特性，通过引入多个线性约束来替代某个非线性约束，但保证转化前后问题是等价的。

constraint formulation of outer approximation

仍记 $f_j(x) = \prod_{i=1}^{m}(1-p_{ij})^{x_{ij}} \quad j \in J$，则对于任意给定的可行域中的点 $\bar{x}$，有

$$\begin{aligned}\nabla f(\bar{x})(x-\bar{x}) &= f(\bar{x})\sum_{i=1}^{m}\ln(1-p_{ij})(x_{ij}-\bar{x_{ij}}) \\ &= f(\bar{x})\sum_{i=1}^{m}\ln(1-p_{ij})x_{ij} - f(\bar{x})\sum_{i=1}^{m}\ln(1-p_{ij})\bar{x_{ij}}\end{aligned}$$

其中前一项是变量的线性组合，有一项在 $\bar{x}$ 给定时是常数。
依据此外逼近约束的形式，即可得到问题的外逼近模型：

outer approximation model

若记问题中的全部整数可行解为集合 X，则模型可以描述为

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} a_j \eta_j \qquad \text{(OA)} \\
\text{s.t.} \quad & \eta_j \geq f(\bar{x}) \sum_{i=1}^{m} \ln(1-p_{ij})(x_{ij} - \bar{x_{ij}}) + f(\bar{x}), \quad \forall\, j \in J,\ \bar{x} \in X \\
& \sum_{i=1}^{m} x_{ij} \leq 1 \quad \forall\, j \in J, \\
& x_{ij} \in \{0,1\} \quad \forall\, j \in J \quad i \in I.
\end{aligned}
$$

可以看出，此时外逼近约束的个数非常多，对问题的求解造成了一定的困难，因此考虑限制后的模型，即近考虑部分约束。

Idea of outer approximation method

基本思路：外逼近约束个数过多，仅考虑其子集 $\hat{X} \subseteq X$

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} a_j \eta_j \\
\text{s.t.} \quad & \eta_j \geq f(\bar{x}) \sum_{i=1}^{m} \ln(1-p_{ij})(x_{ij} - \bar{x_{ij}}) + f(\bar{x}), \quad j \in J, \ \bar{x} \in X \\
& \sum_{i=1}^{m} x_{ij} \leq 1 \quad j \in J, \quad x_{ij} \in \{0,1\} \quad j \in J \quad i \in I
\end{aligned}
$$

1. 去除全部外逼近约束，给出当前最优解 x, η
2. 检验当前最优解是否能满足全部的外逼近约束，若可以，迭代终止，执行结束
3. 否则选择违背最严重的外逼近约束，加入到其中，重新求解并执行步骤 2

choose violate constraint

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} a_j \eta_j \\
\text{s.t.} \quad & \eta_j \geq f(\bar{x}) \sum_{i=1}^{m} \ln(1-p_{ij})(x_{ij} - \bar{x_{ij}}) + f(\bar{x}), \quad j \in J,\ \bar{x} \in X \\
& \sum_{i=1}^{m} x_{ij} \leq 1 \quad j \in J, \quad x_{ij} \in \{0,1\} \quad j \in J \quad i \in I
\end{aligned}
$$

- 除非已经找到了最优解，否则限制后的问题的解一定会带来一个违背的约束（放大了可行域，目标函数一定会更小，说明它不在原问题的可行域当中）
- 可以直观地理解，违背即指 η_j 在可行域之外了，由于本身是求的最小化的问题，因此在找到的点加割即为最违背的割。
- 由此可以得到当求解出一个在可行域当中的点时，如果 $\eta < f_j(x)$，就说明这个点应该被包含在可行域之外，因此可以在这个点加入一个割。

Columnn Generation

Basic Idea

- 许多线性规划问题的列数（变量）太多而无法明确考虑所有变量
- 在算法的开始仅使用部分变量，并假设其他变量全部为 0
- 迭代地将有可能改进目标函数的变量添加到数学规划模型中。一旦可以证明添加新变量将无法再提高目标函数的值，终止迭代过程并得到最优解。

How to use column generation in WTA Problem

- **基本思路**：通过将所有目标场景 S 列出来的方式，将问题转化为一个线性规划问题。
- **转化方式**：假设一共有 m 个武器，则对于任意一个目标 j，每个武器可以选择打击 j 或不打击 j，因此共有 2^m 个不同的打击方案。
- **一个例子**：一共有 8 个武器，则使用第 1，3，6 号武器的攻击方案即记为 $s_{[1,0,1,0,0,1,0,0]}$，$|S| = 2^8 = 256$
- n_{si}：0-1 变量，表示在第 s 个场景下，是否启用武器 i. 在上面的例子中，$n_{s1} = 1$，$n_{s2} = 0$
- $q_{js} = a_j \prod_{i=1}^{m}(1 - n_{si} \cdot p_{ij})$：使用第方案 s 打击目标 j 的概率并加权，比如 q_{3s} 就是采用 1,3,6 号武器打击 3 号目标的击毁概率乘以目标 j 的权重。

transformed formulation

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} q_{js} y_{js} & & \text{(CG)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} n_{si} y_{js} \leq 1 & \forall\ i \in I & \\
& \sum_{s=0}^{2^m-1} y_{js} = 1 & \forall\ j \in J & \\
& y_{js} \in \{0,1\} & \forall\ j \in J,\ s \in S &
\end{aligned}
$$

- $I = \{1, \cdots, m\}$, 武器集合.
- $J = \{1, \cdots, n\}$, 目标集合.
- $S = \{1, \cdots, 2^m\}$, 场景集合.
- n_{si}: 场景 s 是否使用武器 i
- y_{js}: 0-1 变量，是否对目标 j 应用 s
- q_{js}: 赋权后的场景 s 对 j 的摧毁概率

transformed formulation continuous

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} q_{js} y_{js} & & \text{(CG)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} n_{si} y_{js} \le 1 & \forall\, i \in I & \\
& \sum_{s=0}^{2^m-1} y_{js} = 1 & \forall\, j \in J & \\
& y_{js} \in \{0,1\} & \forall\, j \in J,\ s \in S &
\end{aligned}
$$

- **目标函数**：最小化赋权后的消灭概率
- **第一个约束**：每个武器最多只能攻击一个目标
- **第二个约束**：每个目标恰好安排一个场景

column enumeration

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} q_{js} y_{js} && && \text{(CG)} \\ \text{s.t.} \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} n_{si} y_{js} \leq 1 && \forall\ i \in I \\ & \sum_{s=0}^{2^m-1} y_{js} = 1 && \forall\ j \in J \\ & y_{js} \in \{0,1\} && \forall\ j \in J,\ s \in S \end{aligned}$$

- 列枚举方法的基本思路是采用较好的方式对全部列进行枚举
- 文章中采用了两个技巧：weapon number bounding and weapon domination
- weapon number bounding：如果给一个目标分配了太少或者太多武器的场景，一定会导致无法改进。
- weapon domination：针对某个目标，安排某个武器一定比安排另一个武器更好

LP Relaxition

$$
\begin{aligned}
\min \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} q_{js} y_{js} && && \text{(CG-LP)} \\
\text{s.t.} \quad & \sum_{j=1}^{n} \sum_{s=0}^{2^m-1} n_{si} y_{js} \leq 1 && \forall\ i \in I \\
& \sum_{s=0}^{2^m-1} y_{js} = 1 && \forall\ j \in J \\
& y_{js} \geq 0 && \forall\ j \in J,\ s \in S
\end{aligned}
$$

- 将问题进行线性松弛，第三个约束可以直接放成 $y_{js} \geq 0$
- 此时线性规划的特点为包含了 $n \times 2^m$ 列
- 列生成思路：仅取其中的部分列 (变量)，因为在线性规划的最优解中，最多有 $m+n$ 个变量不为 0，即其他变量对应的对偶约束不积极。

Dual Problem

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{m} u_i + \sum_{j=1}^{n} v_j \\
\text{s.t.} \quad & \sum_{i=1}^{m} x_{is} u_i + v_j \le q_{js} \quad \forall\ (s,j) \in X \\
& u_i \le 0, \quad v_j \ \text{free}
\end{aligned}
\tag{CG-Dual}
$$

- 若记 $X = \{(s,j) | s \in S, j \in J\}$，每一个 X 中的元素都对应一个约束。
-
- 选择原问题中的部分列相当于对偶问题选择了部分行。

Restricted Dual Problem

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{m} u_i + \sum_{j=1}^{n} v_j \\
\text{s.t.} \quad & \sum_{i=1}^{m} x_{is} u_i + v_j \le q_{js} \qquad \forall (s,j) \in \hat{X} \\
& u_i \le 0, \quad v_j \ \text{free}
\end{aligned}
\tag{CG-Dual-R}
$$

- 选择部分约束即为仅考虑由部分 (s,j) 生成的约束，记 $\hat{X} \subseteq X$
- 求解得到 u^*, v^* 并带入原对偶，若都成立则一定为最优解。
- 否则希望选择一个违背最严重的，即 $\sum_{i=1}^{m} x_{is} u_i + v_j > q_{js}$ 中最大的。

subproblem

$$\begin{aligned} \min \quad & a_j \prod_{i=1}^{m} (1 - p_{ij} x_{is}) - \sum_{i=1}^{m} x_{is} u_i - v_j \\ \text{s.t.} \quad & j \in J, \quad s \in S \end{aligned} \tag{CG-sub}$$

- 希望选择一个违背最严重的，即 $\sum_{i=1}^{m} x_{is} u_i + v_j > q_{js}$ 中最大的。
- 将 q_{js} 的定义带入，即可得到上述子问题。
- 可以将问题分离，之后对每个给定的目标 j 求解，其直观为使用每个武器攻击需要一个花费，希望平衡开启武器的花费和消灭目标的概率。
- 该子问题的形式的形式可以有效利用之前的外逼近方法。

Future Research Plan

Future Research Plan

- 实现列生成方法
- 在问题中加入传感器

Alternate applications

$$\min \quad \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} (1 - p_{ij})^{x_{ij}} \right) \tag{4}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall\, i \in I, \tag{5}$$

$$x_{ij} \in \{0, 1\} \quad \forall\, j \in J,\ i \in I. \tag{6}$$

- 该模型不仅可以被应用在武器目标分配问题中，一些资源有限的分配问题都可以利用该模型。
- 目标函数可以变化为 $\sum_{j=1}^{n} V_j f_j(x)$，只要它满足一定的条件，比如是凸函数，则可以考虑使用上述两个方法来进行求解。

Acknowledgement and Reference

Thank you!