

文章编号: 2095-2163(2020)10-0071-04

中图分类号: U285

文献标志码: A

# 基于 SCA 铁路安全监测用无人机云计算优化算法

陈明阳

(北京邮电大学 信息与通信学院, 北京 100876)

**摘要:**近年来开始使用无人机对铁路线路、长大桥隧等基础设施及一些维修人员难以进入的区域进行安全监测,特别是在缺乏通信设施的地区,无人机可以为移动用户提供良好的无线通信服务。本文对基于 SCA 的铁路安全监测用无人机云计算算法的优化研究包括:外部限制条件与优化问题的建模和非凸优化问题的优化两部分,采用 CVX 库基本实现了对于优化算法的仿真。研究表明:采用 SCA 算法进行优化,避免了代价极大的全局优化,对于整体算法的性能有一个很大的提升,为无人机在铁路安全监测中应用提供了理论基础。

**关键词:** SCA 算法; 铁路安全监测; 无人机; 非凸优化

## Cloud computing optimization algorithm of UAV for railway safety monitoring based on SCA

CHEN Mingyang

(Department of Construction Management, Beijing University of Posts and Telecommunications, Beijing 100876, China)

**[Abstract]** In recent years, unmanned aerial vehicles (UAVs) have been used to monitor the security of areas that are difficult for maintenance personnel to access, such as railway lines, Bridges and tunnels, etc., especially in areas where there is a lack of communication facilities, UAVs can provide good wireless communication services for mobile users. In this paper, the optimization study of SCA-based UAV cloud computing algorithm for railway safety monitoring includes two parts: modeling of external constraints and optimization problem and optimization of non-convex optimization problem. CVX library is used to basically realize the simulation of optimization algorithm. The research shows that the adoption of SCA algorithm for optimization avoids costly global optimization and greatly improves the performance of the overall algorithm, which provides a theoretical basis for the application of UAVs in railway safety monitoring.

**[Key words]** SCA algorithm; Railway safety monitoring; Unmanned aerial vehicle (UAVs); The convex optimization

### 0 引言

近年来无人机越来越多的应用于铁路线路、长大桥隧等基础设施及一些维修人员难以进入的区域,特别是在通信设施匮乏的地区,应用无人机进行铁路安全监测尤为必要。无人机一方面可用作空中基站,增强无线网络覆盖、容量、可靠性和能效;另一方面,还可用作蜂窝网络内的飞行移动终端。事实上,特别是在通信信号弱的地区,比如川藏铁路的山区、长大铁路隧道中、信号强度弱的区域,无人机辅助通信可以全方位覆盖,为现存通信基础设施提供无空隙服务<sup>[1]</sup>。因此,研究基于 SCA 铁路安全监测用无人机云计算算法具有十分重要的意义,为无人机在铁路安全监测中应用提供理论基础和技术支撑。进一步研究包括优化无人机的飞行能力与路径,结合无人机提供通信信道的特点,挖掘出无人机未来在铁路通信领域的更大潜力尤为重要。事实上,无人机辅助无线通信提供了一种有前途的解决方案,可为没有基础设施覆盖的设备提供无线连接,

因此,基于 SCA 的铁路安全监测用无人机云计算优化算法的研究势在必行。

相对于传统云计算的架构,比如地面云计算网络,无人机云计算结构有更加显著的优势。无人机可以在任意场景下提供云计算服务,比如沙漠、高原或者地理环境比较复杂多变的地方,而这些地方是不能建立地面云计算服务器的。无人机的计算能力要更加优越,并具有更短的时延,因为短距离的 line-of-sight 更容易得到保障。无人机的轨迹可以根据不同任务进行优化,以更好的提高计算性能,而传统网络的可塑性较弱。基于上述优点,国际上许多科研机构已经启动了无人机云计算相关项目的研究,并将其投入使用。

本文拟采取的优化方案是配置一个移动无人机,并向铁路移动设备提供计算卸载服务。无人机云计算问题可以简化为一个优化问题,即在一定边界条件下,比如无人机飞行能力,包括距离、速度等;无人机数据传输能力等一系列条件下,通过优化无

作者简介: 陈明阳(1996-),男,硕士研究生,主要研究方向:无人机动态系统与控制、数据挖掘研究。

收稿日期: 2020-05-20

人机飞行轨迹与传输数据分配,以得到最优解。本文研究的问题是非凸优化问题,通过 Successive Convex Approximation(SCA)算法,将其变为收敛的迭代问题,从而得解。SCA 优化算法是通过连续的迭代,并且保证收敛性,以获得局部最小点。通过将原本非凸优化问题经过一系列近似,变为凸优化问题,再使用 SCA 算法优化。这种算法避免了代价极大的全局优化,对于整体算法的性能有一个很大的提升。

## 1 无人机云计算优化算法

基于 SCA 铁路安全监测用无人机云计算优化算法的研究包括:外部限制条件与优化问题的建模和非凸优化问题的优化两个部分。

### 1.1 外部限制条件与优化问题的建模

对于铁路用铁路安全监测用无人机飞行的限制有几方面,由于无人机飞行技术的限制,导致无人机的飞行速度是有限制的。所以在优化无人机轨迹的同时,要考虑相关速度限制。无人机在第  $n$  个时间间隔内的飞行速度为式(1)<sup>[2]</sup>:

$$v_n^c = \frac{p_{n+1}^c - p_n^c}{\Delta} \quad (1)$$

式中,  $v_n^c$  为飞行速度, m/s;  $p_{n+1}^c$  为在  $c$  点  $n+1$  时刻的运行轨迹, m;  $p_n^c$  为在  $c$  点  $n$  时刻的运行轨迹, m;  $\Delta$  为第  $n$  个时间间隔内的飞行时间, s。

因为上述无人机飞行速度限制,可以得到表达式(2)<sup>[2]</sup>:

$$\|v_n^c\| = \frac{\|p_{n+1}^c - p_n^c\|}{\Delta} \leq v_{max} \quad (2)$$

式中,  $v_{max}$  为最大飞行速度, m/s。

对于无人机总体的飞行,需要在指定的时间  $T$  内完成指定路程,所以可以得到式(3)对于速度的限制<sup>[2]</sup>:

$$\frac{\|p_{N+1}^c - p_1^c\|}{N} \leq v_{max} \quad (3)$$

式中,  $v_n^c$  为飞行速度, m/s;  $p_{N+1}^c$  为在  $c$  点  $N+1$  时刻的运行轨迹, m;  $p_1^c$  为在  $c$  点第一时刻的起始位置, m;  $N$  为第  $N$  个时间间隔内的飞行时间, s。

为了确保无人机可以准时的处理完移动用户发送的数据,对于无人机 CPU 的转数也存在一系列限制。本文考虑在第  $n$  个时间间隔之中无人机本地计算的过程。不妨设此时无人机进行计算的 CPU 的频率是  $f_n^c$ , 需要处理的总的的数据为  $\sum_{k=1}^K l_{k,n}$ 。据此,可以计算推得在第  $n$  个时间间隔内无人机 CPU

频率为式(4)<sup>[2]</sup>:

$$f_n^c = \frac{\sum_{k=1}^K C_k l_{k,n}}{\Delta} \quad (4)$$

式中,  $f_n^c$  为频率, Hz;  $\sum_{k=1}^K C_k l_{k,n}$  为  $n$  个时间间隔内脉冲之和, Hz;  $\Delta$  为时间间隔, s。

继续推导得到计算卸载所需的计算能量为式(5)<sup>[2]</sup>:

$$E_{k,n}^c(l_n) \triangleq E_k^c(l_n, f_n^c) = \frac{\gamma^c C_k l_{k,n}}{\Delta^2} \left( \sum_{k'=1}^K C_{k'} l_{k',n} \right)^2 \quad (5)$$

式中,  $E_{k,n}^c(l_n)$  为计算卸载所需的计算能量, J;  $l_{k,n}$  为第  $n$  个时间间隔内  $K$  个移动用户需要无人机处理的全部数据;  $\sum_{k'=1}^K C_{k'} l_{k',n}$  为从 1 到  $K$  时刻全部能量之和。

为简单起见,本文只假设了 3 个移动用户。无人机总体的能量限制比较简介,只需要在全部  $N$  个时间间隔内,无人机飞行能量,无人机传输信息能量,与无人机本地计算能量的总和小于等于总的能量限制。于是可以得到表达式(6)<sup>[3]</sup>:

$$\sum_{k=1}^K \sum_{n=1}^{N-2} E_{k,n+1}^c(l_{n+1}) + E_{O,k,n+2}^c(L_{k,n+2}^c, p_{n+2}^c) + \sum_{n=1}^N E_{F,n}^c(v_n^c) \leq \varepsilon \quad (6)$$

式中,  $\varepsilon$  为无人机飞行过程中可得到的总能量。

以上为无人机云计算能量限制。对于无人机传输信息的分配,也需要满足一系列限制条件。比如在任一时刻,无人机接收到的数据要大于等于无人机本地计算的数据,因为无人机云计算本地处理的数据来自于移动用户向无人机发送的数据。如果前者大于后者,将会与真实情况矛盾,式(7)。

$$\sum_{i=1}^n l_{k,i+1} \leq \sum_{i=1}^n L_{k,i}^m, k \in K, n \in N-2 \quad (7)$$

对于无人机处理完的数据也有限制。无人机发送给地面移动用户的数据要小于等于无人机 CPU 计算处理之后的数据。同时,几个待优化的传输数据变量也要满足限制条件,式(9)~式(11)<sup>[4]</sup>为:

$$\sum_{i=1}^n L_{k,i+2}^m \leq O_k \sum_{i=1}^n l_{k,i+1}, k \in K, n \in N-2, \quad (8)$$

$$\sum_{n=1}^{N-2} L_{k,n}^m = I_k, k \in K, \quad (9)$$

$$\sum_{n=1}^{N-2} l_{k,n+1} = I_k, k \in K, \quad (10)$$

$$\sum_{n=1}^{N-2} O_k = I_k, k \in K. \quad (11)$$

简而言之,在第  $n+1$  个时间间隔无人机云计

算本地计算的总数据要小于等于在第  $n$  个时间间隔以及之前无人机从移动用户接收到的总数据。在第  $n + 2$  个时间间隔之中无人机向移动用户发送的数据要不大于  $n + 1$  时间间隔之前无人机已经计算处理好的数据。

## 1.2 非凸优化问题的优化

首先建立一个集合  $z = \{z_n\}_{n \in N}$ , 其中  $z_n = (\{L_{k,n}^m\}_{k \in K}, \{l_{k,n}\}_{k \in K}, \{L_{k,n}^c\}_{k \in K}, p_n^c)$ , 是本文需要优化的变量。经过仔细的研究发现铁路用无人机接收数据的能耗  $E_{O,k,n}^m(z_n) = E_{O,k,n}^m(L_{k,n}^m, p_n^c)$ , 这意味着可以根据之前的引理, 将该能量表示成二个非负, 且凸函数的乘积。利用引理, 并结合无人机不同能量函数的表达式, 可以得到函数表达式 (12)、(13) [3]:

$$f_1(L_{k,n}^m) = \frac{N_0 B/K}{g_0} (2^{B/\gamma K} - 1). \quad (12)$$

$$f_2(p_n^c) = (x_n^c - x_k^m)^2 + (y_n^c - y_k^m)^2 + H^2, \quad (13)$$

新引入临时变量  $z_n(v) = (\{L_{k,n}^m(v)\}_{k \in K}, \{l_{k,n}(v)\}_{k \in K}, \{L_{k,n}^c(v)\}_{k \in K}, p_n^c) \in X$ , 这是第  $v$  次迭代变量, 也一定在原函数定义域之内。至此, 可以得出关于能量函数的凸近似, 式 (14) [3], 并且适用于 SCA 算法。

$$\begin{aligned} \bar{E}_{O,k,n}^m(z_n; z_n(v)) &\triangleq \bar{E}_{O,k,n}^m(L_{k,n}^m, p_n^c; L_{k,n}^m(v), p_n^c(v)) = \\ &f_1(L_{k,n}^m) f_2(p_n^c(v)) + f_1(L_{k,n}^m(v)) f_2(p_n^c) + \\ &\frac{\tau_{L_{k,n}^m}}{2} (L_{k,n}^m - L_{k,n}^m(v))^2 + \frac{\tau_{x_n^c}}{2} (x_n^c - x_n^c(v))^2 + \\ &\frac{\tau_{y_n^c}}{2} (y_n^c - y_n^c(v))^2. \end{aligned} \quad (14)$$

同理, 对于无人机能量的限制条件依然可采取相似方法。需要将几个关于能量函数的非凸表达式用引理转换为可以在 SCA 算法中运行的凸函数。首先从比较简单的无人机计算能量入手, 式 (15):

$$E_{k,n}^c(z_n) \triangleq E_{k,n}^c(l_n) = \frac{\gamma^c}{\Delta^2} g(x_1, x_2). \quad (15)$$

$h_1(x_1)$  可以分解函数, 并得出  $h_1(x_1) = l_{k,n}$ 。对于  $h_2(x_2)$ ; 同样可以将原函数通过变换得出  $h_2(x_2) = \left(\sum_{k=1}^K C_k l_{k,n}\right)^2$ 。无人机下行传输能量表达式相对复杂, 式 (16):

$$E_{O,k,n}^c(z_n) = E_{O,k,n}^c(L_{k,n}^c, p_n^c) = \frac{N_0 B \Delta / K}{g_0} g(x_1, x_2). \quad (16)$$

对于  $h_1(x_1)$ , 可以通过定义算得  $h_1(x_1) = \frac{l_{k,n}^c}{2^{B/\gamma K}}$ ; 同理对于  $h_2(x_2)$ , 可以发现  $h_2(x_2) =$

$(x_n^c - x_k^m)^2 + (y_n^c - y_k^m)^2 + H^2$ 。之后, 对于第  $v$  次迭代量  $z_n(v)$ , 可以使其满足对于无人机能量限制的函数, 并且可以满足该限制函数的凸上限。这意味着可以继续使用 SCA 算法解决本文所建立的无人机计算卸载优化模型。在该算法中, 引用了一个步长函数  $\{\gamma(v)\}$ , 该函数定义如下: 满足  $\gamma(v) \in (0, 1]$ , 并且  $\gamma(v) \rightarrow 0$ ,  $\sum_v \gamma(v) = \infty$ 。根据对于该函数定义, 本文对函数进行了如下选择:  $\gamma^v = \gamma^{v-1} (1 - 10^{-3} \gamma^{v-1})$ 。可以肯定的是, 此步长可令算法性能有所裨益。

## 2 基于正交方式优化算法的实现

在人工智能相关领域的研究, Python 被经常用来处理一些凸优化问题, 本文采用 MATLAB 编写仿真程序, 使用外部库 CVX 作为辅助工具, CVX 主要是为了解决 disciplined convex programs (DCPs)。所谓 DCP 问题, 就是被转化后的凸优化问题。将问题转化为凸优化问题要满足一系列规则, 如果不满足规则, 程序将会报错。CVX 也可以解决更加复杂的凸优化问题, 其中包括很多不可导问题, 比如函数的范式。CVX 的应用必须确保问题已经变为凸优化问题, CVX 本身并不能转化非凸函数。如果对非凸函数采用 CVX 库将会导致程序报错。

本文对于无人机计算算法优化方案是基于正交方式的信息传输。通过采用 CVX 库基本实现了对于优化算法的仿真。算法的伪代码如下:

先将待优化变量初始化, 猜测一下轨迹, 以及数据分配方式。不妨定义为  $z(0) = \{z_n(0)\}$ ,  $z_n(0) \triangleq (\{L_{k,n}^m(0)\}, \{l_{k,n}(0)\}, \{L_{k,n}^c(0)\}, \{p_n^c(0)\})$ , 并且满足  $\tau_{L_{k,n}^m}, \tau_{x_n^c}, \tau_{y_n^c} > 0$ 。设迭代次数  $v = 0$ ;

(1) for  $v$  in 0: iteration; 其中 iteration 为设定的迭代次数;

(2) If  $z(v)$  是原优化问题最优解, break;

(3) else 根据  $z(v)$  以及转换后的凸优化问题计算  $\hat{z}(z(v))$ ;

(4) 更新  $z(v)$ ,  $z(v+1) = z(v) + \gamma(v) (\hat{z}(z(v)) - z(v))$ ; 其中  $\gamma(v)$  是步长函数;

(5)  $v = v + 1$ ;

输出:  $\{L_{k,n}^m\}, \{l_{k,n}\}, \{L_{k,n}^c\}, \{p_n^c\}$ 。

为使用程序的简要结构。在伪代码的第三步中, 涉及到转换后的凸优化问题。该问题是根据原始模型转换而来。待优化问题如式 (17):

$$\text{minimize } \sum_{k=1}^K \sum_{n=1}^{N-2} \bar{E}_{O,k,n}^m(z_n; z_n(v)). \quad (17)$$

(下转第 79 页)