改进的测量设备无关协议参数优化方法

周江平 周媛媛† 周学军

(海军工程大学电子工程学院, 武汉 430000)

(2023 年 2 月 12 日收到; 2023 年 4 月 19 日收到修改稿)

实际量子密钥分发中参数的优化选择能大幅提升系统密钥生成率和最大传输距离, 由于全局搜索算法的成本过大, 本地搜索算法被广泛地应用. 然而该算法存在两个问题, 一是所得解不一定为全局最优解, 二是算法的有效性极大地受制于初始值的选择. 利用蒙特卡罗方法对密钥生成率函数是否为凸函数进行了证明, 并仿真分析了密钥生成率函数在不同参数维度上的特性, 提出了粒子群本地搜索算法并与本地搜索算法进行仿真比较. 结果表明, 密钥生成率函数为非凸函数, 但合理设置初始值, 本地搜索算法仍能求得全局最优解; 在传输距离较远时, 本地搜索算法因难以通过随机取值的方法得到有效的初始值而失效, 粒子群本地搜索算法能克服这一缺点, 以轻微增加算法复杂度为代价, 提升了系统的最大传输距离.

关键词: 量子密钥分发, 粒子群算法, 测量设备无关协议, 参数优化

PACS: 03.67.Dd, 03.67.Ac, 03.67.Hk, 42.50.Ex **DOI:** 10.7498/aps.72.20230179

1 引 言

量子密钥分发 (quantum key distribution, QKD)[1,2] 以量子物理原理为基础, 可实现远距离双方无条件安全通信. 由于现实中设备的非理想性, 如系统常用弱相干态光源代替理想单光子源, 所用检测器检测能力有限等, 实际 QKD 系统存在诸多易遭受攻击的漏洞 [3–5]. 针对光源有光子数分离攻击 [6], 针对检测设备有致盲攻击、时移攻击、死时间攻击等 [7]. 为此诱骗态协议 [8]、测量设备无关 (measurement device independent, MDI) 协议 [9] 等相继被提出. 二者的结合能得到较好的安全性和较高的密钥生成率, 是一种经典的量子密钥分发协议, 得到了广泛的研究和应用 [10,11].

在量子密钥分发实际应用中, 通过优化相关参数的选择可有效地提升系统实际性能. 然而, 实际系统中可调参数多, 优化难度大. 早期优化主要依靠理论分析, 仅对部分参数进行优化, 如 Lo 等 [12] 针对 BB84 协议优化了 X 基和 Z 基的选择概率, 从而提升了系统性能; 随着计算机的不断发展, 人们对主要参数 (如信号强度等) 进行全局搜索优化 [13], 受限于全局搜索的巨大成本, 只能凭经验固定部分参数 (如基选概率等) 从而降低搜索的维度, 在一定程度上提升了系统性能; 2014 年, Xu 等 [14] 提出了基于本地搜索算法 (local search algorithm, LSA) 的全局优化方法, 首次将所有参数统筹优化, 极大提升了系统性能; 此后参数优化的研究主要集中于针对时变系统的快速参数优化, 大多先以 LSA 算法求解出大量不同环境下的最优参数, 再以此数据为基础, 训练出合适的机器学习模型用对新环境下最优参数进行预测 [15–19].

参数的优化不仅对系统性能有重大影响, 也直接关系到基于机器学习参数优化方法的性能. 虽然 LSA 算法速度相对较快, 但是针对非凸目标函数容易陷入局部最优解 [20], 只有在特定情况, 局部最优才为全局最优. Xu 等 [14] 已经用三维图像的形式对密钥生成率函数为凸函数进行了证明, 但这

† 通信作者. E-mail: EPJZYY@aliyun.com

种证明不够严格. 相反, 本文采用蒙特卡罗方法证明了密钥生成率函数并非凸函数. 因而 LSA 算法所得结果是否为最优有待进一步证明, 同时, LSA 算法受初始值影响大, 在系统衰减较大时, 随机得到的有效初始值概率很小, 有必要寻找一种新的方法来寻找全局最优解. 本文引入改进的粒子群算法 (particle swarm optimization, PSO)[21,22], 提出一种 PSO+LSA 的全局参数优化方法 PSLSA, 增强了全局寻优能力, 减少了参数优化对初始值的依赖, 增强了系统在衰减较大时参数的优化能力, 提升了系统性能. 该方法同样适用于采用其他量子密钥分发协议[23–27]的 QKD 系统.

本文首先讨论了密钥生成率函数及凸函数判别方法, 详细描述 PSLSA 算法; 随后通过仿真证明密钥生成率函数为非凸函数, 对 LSA 算法的全局寻优能力、初始值敏感性进行仿真分析; 最后对 PSLSA 算法与 LSA 算法的全局寻优性能进行比较分析.

2 考虑统计波动的诱骗态 MDI 协议

2.1 诱骗态 MDI 协议

诱骗态 MDI 协议具体过程如下:

1)Alice 随机选取 $\{0,1\}$作为密钥比特, 以固定的概率$\{P_\mu,P_\nu,P_\omega\}$随机选取信源强度$\{\mu,\nu,\omega\}$, 以条件概率$\{P_{\mathrm{X}|\mu},P_{\mathrm{X}|\nu},P_{\mathrm{X}|\omega}\}$($\{1-P_{\mathrm{X}|\mu},1-P_{\mathrm{X}|\nu},1-P_{\mathrm{X}|\omega}\}$) 随机选择 X(Z) 基作为编码基进行编码, 最后添加随机相位以制备相干态, Bob 执行相同操作.

2)Alice 和 Bob 分别将光脉冲发送至不可信第三方 Charlie, Charlie 执行检测之后公开声明检测结果. Alice 和 Bob 仅保留所选基一致且相应 Charlie 检测结果有效的比特.

3) 重复执行 1) 和 2) 足够次数后, 假设 Alice 和 Bob 的信源强度分别为$q_{\mathrm{a}},q_{\mathrm{b}}$, 所选基$\lambda\in\{\mathrm{X},\mathrm{Z}\}$, 根据有限的数据测量得到错误率$E^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}$和增益$Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}$.

4) Alice 和 Bob 对双方都发送 1 个光子时接收端有效响应的计数率Y^{Z}_{11}和错误计数率e^{X}_{11}进行估计, 随后经过纠错、私密放大等操作得到最终密钥生成率.

MDI 协议的密钥生成率公式为[9]

$$R\geqslant P^{\mathrm{Z}}_{11}Y^{\mathrm{Z}}_{11}\left[1-H\left(e^{\mathrm{X}}_{11}\right)\right]-Q^{\mathrm{Z}}_{\mu\mu}f_{\mathrm{e}}H\left(E^{\mathrm{Z}}_{\mu\mu}\right),\quad(1)$$

其中, 下标 11 表示通信双方均发送 1 个光子, $\mu\mu$表示通信双方信号强度均为μ; 上标 Z 表示通信双方都选择 Z 基, 相应的 X 表示通信双方选择 X 基; P^{Z}_{11}为信源产生光子态的概率, 若光源为相干态光源$P^{\mathrm{Z}}_{11}=P^2_{\mu}\left(1-P_{\mathrm{X}|\mu}\right)^2\mu^2\exp\left(-2\mu\right)$; $Q^{\mathrm{Z}}_{\mu\mu}$为总增益; $H\left(x\right)$为熵函数, 其表达式为$H\left(x\right)=-x\log_2\left(x\right)-\left(1-x\right)\log_2\left(1-x\right)$; f_{e}为纠错效率, 一般取常数; $E^{\mathrm{Z}}_{\mu\mu}$表示量子比特误码率.

在诱骗态 MDI 协议中, 通过测量不同信源强度下的总增益和量子比特误码率, 可以估计出Y^{Z}_{11}和e^{X}_{11}, 从而得到最终密钥生成率[28,29]. 考虑实际情况下有限长数据效应, 实验值与理论值之间存在偏差, 因而需对估计公式进行相应修正. 以典型的标准差分析法[30]来进行统计波动分析, 实验值会以一定概率处于某一区间, 分别用上标 U 和 L 表示根据测量值估计得到的区间的上界和下界. 估计Y^{Z}_{11}和e^{X}_{11}的约束条件可以写成如下形式[28]:

$$\begin{aligned}
&Q^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}\leqslant\sum_{n,m=0}\exp\left(-q_{\mathrm{a}}-q_{\mathrm{b}}\right)\frac{q^n_{\mathrm{a}}}{n!}\frac{q^m_{\mathrm{b}}}{m!}Y^{\lambda}_{nm}\leqslant Q^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}},\\
&E^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}\leqslant\sum_{n,m=0}\exp\left(-q_{\mathrm{a}}-q_{\mathrm{b}}\right)\frac{q^n_{\mathrm{a}}}{n!}\frac{q^m_{\mathrm{b}}}{m!}e^{\lambda}_{nm}Y^{\lambda}_{nm}\\
&\qquad\leqslant E^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}},\\
&Q^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}=Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}\left(1+\beta_q\right);\quad Q^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}=Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}\left(1-\beta_q\right),\\
&E^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}=E^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}\left(1+\beta_{eq}\right);\\
&E^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}=E^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}\left(1-\beta_{eq}\right).
\end{aligned}\quad(2)$$

与 (1) 式中定义类似, $Y^{\lambda}_{nm},e^{\lambda}_{nm}$分别表示计数率和错误计数率, $Q^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}$和$Q^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}$($E^{\lambda,\mathrm{U}}_{q_{\mathrm{a}}q_{\mathrm{b}}}$和$E^{\lambda,\mathrm{L}}_{q_{\mathrm{a}}q_{\mathrm{b}}}$) 分别为$Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}$($E^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}$) 的上界和下界. 波动率$\beta_q,\beta_{eq}$可表示为

$$\begin{aligned}
&\beta_q=\min\left(\frac{n_\alpha}{\sqrt{N^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}}},1\right),\\
&\beta_{eq}=\min\left(\frac{n_\alpha}{\sqrt{N^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}E^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}Q^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}}},1\right).
\end{aligned}\quad(3)$$

其中n_α为标准差, $N^{\lambda}_{q_{\mathrm{a}}q_{\mathrm{b}}}$为选择对应的信源强度和基底时, 发送的总脉冲个数.

二诱骗态情况下, 根据 (2) 式, Y^{Z}_{11}和e^{X}_{11}可用线性规划的方法来估计, 还可以假设信源强度满足$\mu>\nu>\omega\geqslant0$, 用下式进行估计[31,32]:

$$
\begin{aligned}
Y_{11}^{\mathrm{Z}} \geqslant Y_{11}^{\mathrm{Z,L}} = & \frac{1}{(\mu-\omega)^2(\nu-\omega)^2(\mu-\nu)} \\
& \times \Big[\left(\mu^2-\omega^2\right)(\mu-\omega)\left(Q_{\nu\nu}^{\mathrm{Z,L}} \exp(2\nu) + Q_{\omega\omega}^{\mathrm{Z,L}} \exp(2\omega) - Q_{\nu\omega}^{\mathrm{Z,U}} \exp(\nu+\omega) - Q_{\omega\nu}^{\mathrm{Z,U}} \exp(\omega+\nu)\right) \\
& - \left(\nu^2-\omega^2\right)(\nu-\omega)\left(Q_{\mu\mu}^{\mathrm{Z,U}} \exp(2\mu) + Q_{\omega\omega}^{\mathrm{Z,U}} \exp(2\omega) - Q_{\mu\omega}^{\mathrm{Z,L}} \exp(\mu+\omega) - Q_{\omega\mu}^{\mathrm{Z,L}} \exp(\omega+\mu)\right) \Big], \\
e_{11}^{\mathrm{X}} \leqslant e_{11}^{\mathrm{X,U}} = & \frac{1}{(\nu-\omega)^2 Y_{11}^{\mathrm{X,L}}} \left[E_{\nu\nu}^{\mathrm{X,U}} Q_{\nu\nu}^{\mathrm{X,U}} \exp(2\nu) + E_{\omega\omega}^{\mathrm{X,U}} Q_{\omega\omega}^{\mathrm{X,U}} \exp(2\omega) \right. \\
& \left. - E_{\nu\omega}^{\mathrm{X,L}} Q_{\nu\omega}^{\mathrm{X,L}} \exp(\nu+\omega) - E_{\omega\nu}^{\mathrm{X,L}} Q_{\omega\nu}^{\mathrm{X,L}} \exp(\omega+\nu) \right], \qquad (4)
\end{aligned}
$$

式中, $Y_{11}^{\mathrm{X,L}}$ 的估计公式与 $Y_{11}^{\mathrm{Z,L}}$ 相似, 仅仅只有基的差别.

综合 (1)—(4) 式可以看出, 密钥生成率与如下参数的选择有关: 信号态、诱骗态信源强度 μ,ν,ω, 选择信号态、诱骗态的概率 P_μ, P_ν, 选择 ω 的概率 $P_\omega = 1-P_\mu-P_\nu$, 以及确定信源强度后选择 X 基的条件概率 $P_{\mathrm{X}|\mu}, P_{\mathrm{X}|\nu}, P_{\mathrm{X}|\omega}$, 选择 Z 基的条件概率 $1-P_{\mathrm{X}|\mu}, 1-P_{\mathrm{X}|\nu}, 1-P_{\mathrm{X}|\omega}$, 因而独立参数只有 8 个. 将相关参数组合得到 $\boldsymbol{x} = [\mu,\nu,\omega,P_\mu,P_\nu,P_{\mathrm{X}|\mu},P_{\mathrm{X}|\nu},P_{\mathrm{X}|\omega}]$ 为参数选择向量, 本文的问题即为寻找最优的 $\boldsymbol{x}$ 使得密钥生成率 R 最大.

2.2 密钥生成率函数的非凸性判别

凸函数的定义: 定义域为凸集 $\boldsymbol{\Omega}$ 的函数, 对任意的 $\boldsymbol{x},\boldsymbol{y}\in\boldsymbol{\Omega}$, $0\leqslant\theta\leqslant1$, 满足:

$$
f(\theta\boldsymbol{x}+(1-\theta)\boldsymbol{y}) \leqslant \theta f(\boldsymbol{x}) + (1-\theta) f(\boldsymbol{y}). \qquad (5)
$$

基于此定义, 构造密钥生成率函数凸性判别函数:

$$
\begin{aligned}
F(\theta,\boldsymbol{x}_1,\boldsymbol{x}_2) = & R(\theta\boldsymbol{x}_1+(1-\theta)\boldsymbol{x}_2) - \theta R(\boldsymbol{x}_1) \\
& - (1-\theta) R(\boldsymbol{x}_2), \qquad (6)
\end{aligned}
$$

其中 θ 为 $[0,1]$ 中任意常数, $\boldsymbol{x}_1,\boldsymbol{x}_2$ 为两组不同参数, 分别称为两个点.

当 $F\geqslant0$ 恒成立或 $F\leqslant0$ 恒成立时, 密钥生成率函数为凸函数, 反之则为非凸函数.

3 参数优化方法

待优化选择的参数具有 8 个维度, 假设在每一个维度定义域内均匀取样 100 个点, 那么全局搜索需要迭代的次数为 10^{16}, 如果利用普通办公电脑, 所需要的时间超过 57 年, 这是一项不可能完成的任务. Xu 等 [14] 结合坐标下降算法和线性回溯算法 [20] 提出 LSA 算法, 将迭代次数减少到 10^2 量级, 在较短的时间内寻找到了局部最优解, 大幅提升了系统性能.

LSA 算法的具体流程如下:

目标函数 $f_{\mathrm{LSA}}(x_1,\cdots,x_n)$

根据经验初始化搜索位置 $p_0=(x_1^0,\cdots,x_n^0)$, 初始化迭代次数 $t=0$;
while (迭代判决条件) do
  for $k=1{:}n$
    对 $f_{\mathrm{LSA}}\left(x_1^{t+1},\cdots,x_{k-1}^{t+1},x_k^t,x_{k+1}^t,\cdots,x_n^t\right)$ 在 x_k^t 维度上进行线性回溯搜索
    if $f_{\mathrm{LSA}}\left(x_1^{t+1},\cdots,x_{k-1}^{t+1},x_k^{t+1},x_{k+1}^t,\cdots,x_n^t\right) >$ $f_{\mathrm{LSA}}\left(x_1^{t+1},\cdots,x_{k-1}^{t+1},x_k^t,x_{k+1}^t,\cdots,x_n^t\right)$
      更新 x_k^t 为 x_k^{t+1}
      $p_{t+1}=\left(x_1^{t+1},\cdots,x_{k-1}^{t+1},x_k^{t+1},x_{k+1}^t,\cdots,x_n^t\right)$
    else
      $p_{t+1}=\left(x_1^{t+1},\cdots,x_{k-1}^{t+1},x_k^t,x_{k+1}^t,\cdots,x_n^t\right)$
    end
  end
  更新迭代次数 $t=t+1$
  更新搜索位置为 p_{t+1}
end
输出最终结果 $\left(x_1^{t+1},\cdots,x_n^{t+1}\right)$ 和 $f_{\mathrm{LSA}}\left(x_1^{t+1},\cdots,x_n^{t+1}\right)$

然而, LSA 算法在目标函数为非凸时局部最优解为全局最优解还需进一步证明, 并且其对初始位置十分敏感, 针对本文所讨论的密钥生成率问题来看, 其在有限长数据情况下, 优化所得结果还有较大的提升空间. 因此本文提出 PSLSA 优化方法改善 LSA 算法的初值选择.

粒子群算法是一种受自然界中群体智能行为启发的智能优化算法, 因其简单、灵活、高效, 在全局优化问题中得到了广泛地应用 [33,34]. 粒子群算法中, 可根据经验设定粒子的初始位置, 在算法中加入经验知识, 有助于提升算法的性能, 对剩余粒子随机赋予初始位置, 可增强算法的全局搜索能力 [33],

速度更新策略对粒子群算法的全局寻优能力、收敛速度、算法精度等都有影响, 本文采用改进的二阶振荡粒子群算法[22]速度更新策略, 优化算法性能, 进化方程如下:

$$\begin{aligned}
&\boldsymbol{\nu}_i^{t+1}=\kappa\boldsymbol{\nu}_i^t+\varphi_1\left[\boldsymbol{x}_i^*-(1+\varepsilon_1)\,\boldsymbol{x}_i^t+\varepsilon_2\boldsymbol{x}_i^{t-1}\right]+\varphi_2\left[\boldsymbol{g}_{\rm a}-(1+\varepsilon_3)\,\boldsymbol{x}_i^t+\varepsilon_4\boldsymbol{x}_i^{t-1}\right],\\
&\begin{cases}
0<\varepsilon_2<\dfrac{1+\varepsilon_1}{2},\ 0<\varepsilon_4<\dfrac{1+\varepsilon_3}{2},\ 0<\varepsilon_1<1,\ 0<\varepsilon_2<1, & i<M/2,\\
\varepsilon_1<\varepsilon_2-1,\ \varepsilon_3<\varepsilon_4-1,\ 0<\varepsilon_2<1,\ 0<\varepsilon_4<1, & i>M/2,
\end{cases}\\
&\boldsymbol{x}_i^{t+1}=\boldsymbol{x}_i^t+\boldsymbol{v}_i^{t+1},
\end{aligned}\tag{7}$$

式中$\boldsymbol{\nu}_i^t$表示t时刻第i个粒子的速度, κ为惯性权重, φ_1,φ_2为学习因子与$[0,1]$区间均匀分布随机参数的积, $\varepsilon_1,\varepsilon_2,\varepsilon_3,\varepsilon_4$为满足上述约束条件的参数, $\boldsymbol{x}_i^t$表示t时刻第i个粒子的位置, $\boldsymbol{x}_i^*$表示该粒子的历史最优位置, $g_{\rm a}$表示全局最优位置, M表示总的迭代次数.

本文首先用改进的 PSO 算法得到局部最优, 以该结果作为 LSA 算法的初始位置, 一方面避免了 LSA 算法的随机初始位置无法得到有效解的问题, 另一方面, 给出了较优的初始位置, 减少 LSA 的迭代次数. 算法具体流程如下:

目标函数 $f(\boldsymbol{x})$

根据经验初始化粒子1的位置 $\boldsymbol{x}_1$
初始化剩余 $n-1$ 个粒子的位置 $\boldsymbol{x}_i$ 和所有粒子的速度 $\boldsymbol{\nu}_i$
寻找全局最优解 $\boldsymbol{g}_{\rm a}$, $\boldsymbol{g}_{\rm a}=\min\{f(\boldsymbol{x}_1),\cdots,f(\boldsymbol{x}_n)\}$
while(迭代判决条件) do
  for 所有的粒子 do
    更新速度 $\boldsymbol{\nu}_i^{t+1}$
    更新位置 $\boldsymbol{x}_i^{t+1}$
    计算目标函数在新位置的值
    更新当前粒子的历史最优位置 $\boldsymbol{x}_i^*$
  end
更新全局最优解 $\boldsymbol{g}_{\rm a}$
  end
输出最终结果 $\boldsymbol{x}_i^*$ 和 $\boldsymbol{g}_{\rm a}$
以 $\boldsymbol{g}_{\rm a}$ 为初始点, 采用LSA算法求局部最优解

4 仿真及分析

仿真参数与文献 [14] 保持一致, 主要来源于相关科学实验数据[35]. 具体如表 1 所列. 表中$\eta_{\rm d}$为检测器的检测效率, $e_{\rm d}$为非对准错误率, Y_0为背景计数率, $f_{\rm e}$为纠错效率, χ为错误概率, N为发送的总脉冲数, α为光纤衰减系数.

表 1 用于仿真分析的相关参数

Table 1. Practical parameters for numerical simulations.

$\eta_{\rm d}$/%	$e_{\rm d}$/%	Y_0	$f_{\rm e}$	χ	N	α
14.5	1.5	6.02×10^{-6}	1.16	10^{-7}	10^{12}	0.21

4.1 密钥生成率函数的凸性判别

采用蒙特卡罗方法, 以 1 km 处密钥生成率为例, 在定义域内按照均匀分布随机选取 10000 组$[\theta,\boldsymbol{x}_1,\boldsymbol{x}_2]$, 计算$F$值, 过滤掉$R(\boldsymbol{x}_1)=0$和$R(\boldsymbol{x}_2)=0$的点, 所得图像如图 1 所示.

图 1 分别用红色、蓝色来代表$F(\theta,\boldsymbol{x}_1,\boldsymbol{x}_2)$大于 0、小于 0 两种情况. 结果显示, 无论利用线性规划还是 (4) 式来估计$Y_{11}^{\rm Z}$和$e_{11}^{\rm X}$, 在这 10000 个点的取值中, 凸性判别函数同时存在大于 0 和小于 0 的情况. 具体存在以下 4 个点 (表 2). 从表 2 可以更清晰地看出, 密钥生成率函数不满足凸函数的定义, 故密钥生成率函数是非凸函数.

表 2 可判别密钥生成率非凸的四个点

Table 2. Four points which can be used to discriminate the key rate is non convex function.

参数	$\boldsymbol{x}_1$	$\boldsymbol{x}_2$	$\boldsymbol{x}_3$	$\boldsymbol{x}_4$
μ	0.40	0.60	0.50	0.075
ν	0.037	0.045	0.029	5.9×10^{-3}
ω	7.9×10^{-4}	2.6×10^{-3}	2.0×10^{-3}	2.57×10^{-4}
P_μ	0.15	0.48	0.54	0.020
P_ν	0.18	0.13	0.26	0.48
$P_{{\rm X}\mid\mu}$	0.8	0.14	0.76	0.14
$P_{{\rm X}\mid\nu}$	0.92	0.56	0.89	0.64
$P_{{\rm X}\mid\omega}$	0.25	0.99	0.54	0.55
θ	0.47		0.64	
$F(\theta,\boldsymbol{x},\boldsymbol{y})$	-5.50×10^{-6}		3.84×10^{-6}	

4.2 LSA 算法的全局最优性

分别用表 2 中 4 个点作为初始值, 利用 LSA

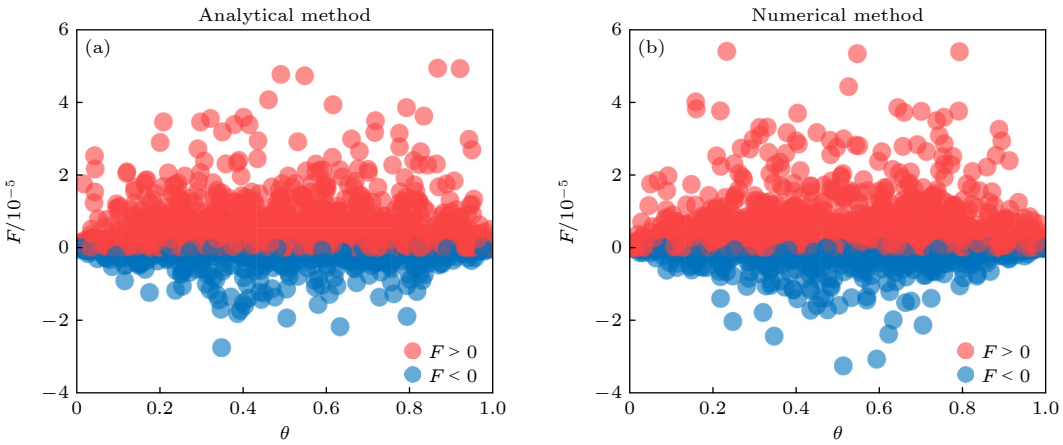

图 1 凸性判别函数 $F(\theta, \boldsymbol{x}_1, \boldsymbol{x}_2)$在 10000 个随机输入下的取值

Fig. 1. Values of convex discriminant function of $F(\theta, \boldsymbol{x}_1, \boldsymbol{x}_2)$ with 10 thousand random input variables.

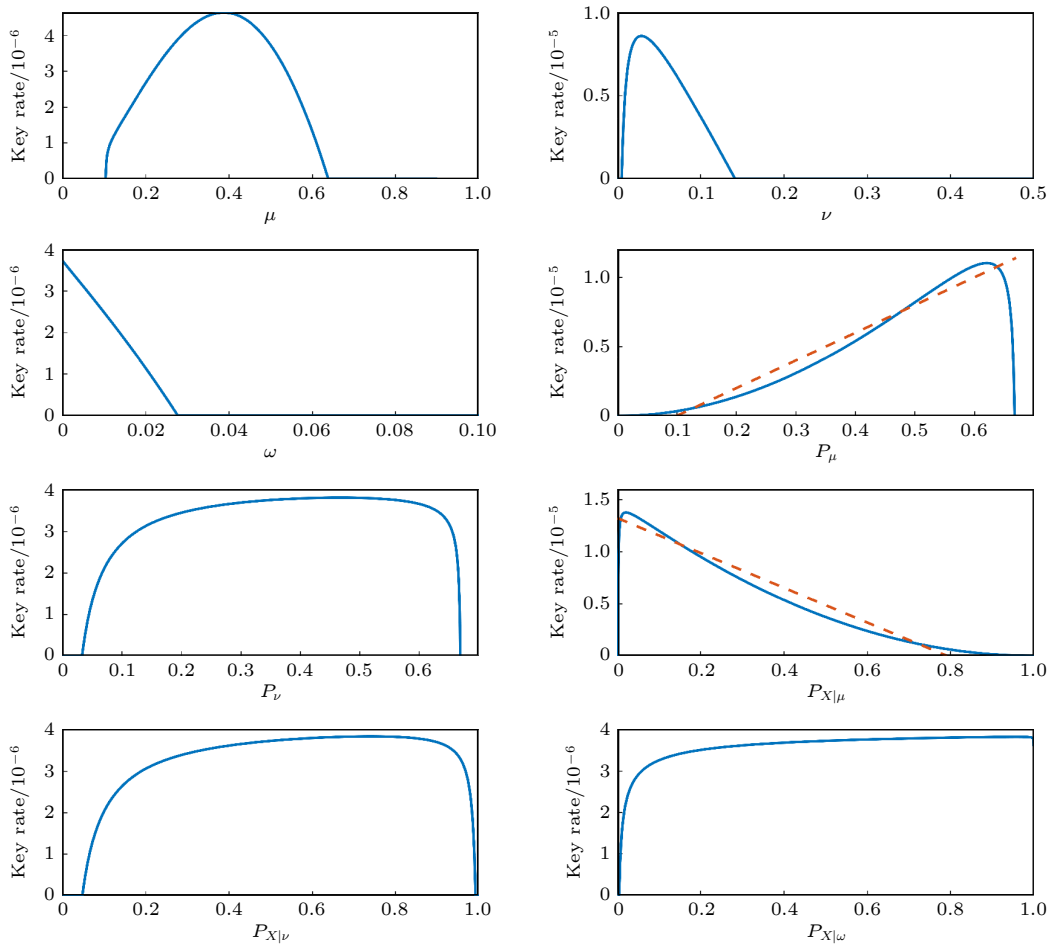

图 2 密钥生成率随单个参数的变化曲线

Fig. 2. Curves of key rate versus each parameter of the input.

算法进行优化得到的结果相同, 均为$[0.46, 0.041, 0, 0.79, 0.15, 0.0050, 0.74, 0.92]$. 这说明尽管密钥生成率函数不是凸函数, LSA 算法仍能收敛到全局最优.

分析 1 km 处, 密钥生成率函数在经验点$[0.5, 0.1, 0, 0.33, 0.33, 0.5, 0.5, 0.5]$处不同维度上投影情况如图 2 所示.

由图 2 可知, 就经验点处而言, 不考虑密钥生成率为 0 的情况, 密钥生成率函数在$P_\mu, P_{X|\mu}$上的投影为非凸函数, 在其余维度上的投影均为凸函数, 并且在非凸情况下, 密钥生成率函数只有一个

驻点, 此时利用 LSA 算法不会陷入局部最优, 所得结果仍为全局最优. 这一方面证明了密钥生成率函数的非凸性, 也说明了 LSA 算法利用合适的初始值, 可以收敛到全局最优.

4.3 LSA 算法对初始值的依赖

LSA 算法对初始值具有较强的依赖性, 首先对不同初始值情况下, 经 LSA 算法优化后密钥生成率性能进行比较分析. 根据文献 [14] 中所列的经验值, 分别设定初始值为

$$\boldsymbol{x}_1 = [0.5, 0.1, 0, 0.33, 0.33, 0.5, 0.5, 0.5],$$

$$\boldsymbol{x}_2 = [0.21, 0.06, 0, 0.33, 0.33, 0.5, 0.5, 0.5],$$

$$\boldsymbol{x}_3 = \left[0.25, 0.05, 10^{-6}, 0.58, 0.30, 0.03, 0.71, 0.83\right].$$

所得密钥生成率如图 3 所示.

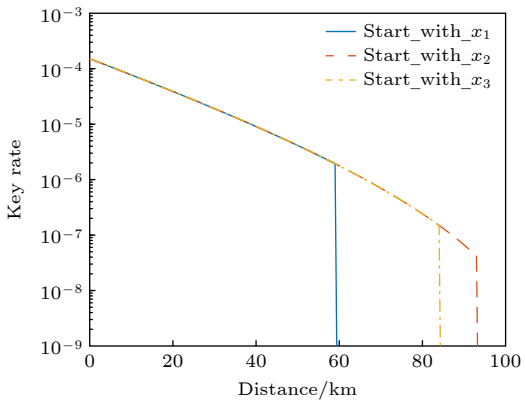

图 3 LSA 优化时不同初始点对密钥生成率的影响

Fig. 3. Influence of different start point on key rate using LSA optimization.

由图 3 可以看出, 比较小于 60 km 处的图像可知, 对于某一确定的距离, 若初始点有效, 则经 LSA 算法优化得到的密钥生成率收敛于相同值, 这说明 LSA 算法能够得到相对较优的解, 具有一定的有效性; 然而, 图像中突然的截断性说明当初始点失效, LSA 算法无法得到合适的参数, 从而得到较好的密钥生成率性能, 当大于 60 km 时, 以 $\boldsymbol{x}_1$ 为初始点无法得到有效的密钥生成率, 当大于 84 km 时, 以 $\boldsymbol{x}_1, \boldsymbol{x}_2$ 为初始点均无法得到有效的密钥生成率. 当系统传输距离较远, 衰减较大时, 在各参数的合理取值范围内, 随机生成 10^5 个点, 分别计算各点的密钥生成率, 仅有 45 个点能够得到非零密钥生成率, 因此可以认为通过随机取值的方法, 得到系统非零初始值概率为 4.5×10^{-4}. 因而, 需要找到一种方法寻找 LSA 算法的有效初始值.

4.4 不同优化算法下二诱骗态 MDI 协议性能比较

分别用 LSA 算法、PSO 算法和本文所提出的 PSLSA 算法对数据长度 $N = 10^{12}$ 情况下二诱骗态 MDI 协议进行参数优化, 密钥生成率性能如图 4 所示.

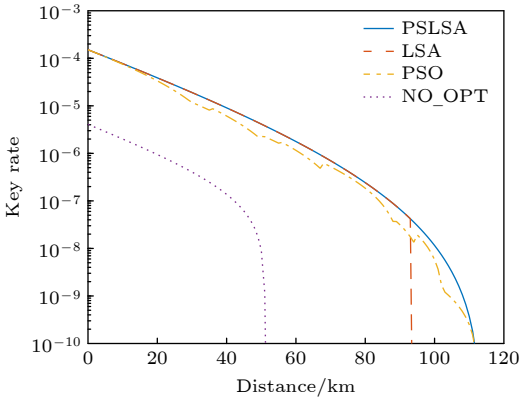

图 4 利用不同优化算法所得密钥生成率对比

Fig. 4. Comparison among three key rates obtained by different optimization algorithm.

图 4 中实线表示利用本文提出的 PSLSA 算法优化后所得密钥生成率, 虚线表示利用 LSA 算法优化, 点划线表示利用 PSO 算法优化, 点线表示未经优化仅凭经验设置参数. 经过优化后, 密钥生成率和最大传输距离均有大幅提升, 这说明对参数的优化选择可极大提升系统性能.

相较 PSO 算法, LSA 算法在初始值合适的情况下, 能够明显提升密钥生成率, 但在距离大于 93 km 时, 由于对初始值的强依赖, 难以得到有效的优化参数, 此时 PSO 算法反而更加有效. 就密钥生成率函数而言, PSO 算法经过足够多的迭代次数, 能与 LSA 算法一样收敛到全局最优, 但是, PSO 算法的全局寻优效率低于 LSA 算法, 迭代次数的增加会显著增加系统成本, 对于可以用 LSA 算法的情景, 没有必要利用 PSO 算法来寻找全局最优.

对于 50 km 处, 不同算法迭代次数、所花时间及最终密钥生成率如表 3 所列. 可以看出, 三者消耗的资源相差不大, 都难以满足时效性要求高的应用场景. 若想解决时效性问题, 还需借助机器学习等方法, 基于 PSLSA 算法优化所得数据对相应机器学习模型进行训练, 可以得到时效性高、功耗小的预测模型. 但是横向比较 PSO, LSA 和 PSLSA,

PSLSA 仍然具有更好的综合性能, 是完成时效性要求不高任务如生成训练数据等的首选方法.

表 3 三种优化算法计算资源消耗比较

Table 3. Comparison of computational resource consumption among the three optimization algorithms.

算法	迭代次数	时间/s	密钥生成率
LSA	858	0.12	4.13×10^{-6}
PSO	40000	4.2	2.97×10^{-6}
PSLSA	40559	4.29	4.13×10^{-6}

本文提出的 PSLSA 算法结合了两种算法的优点, 在密钥生成率上与 LSA 算法相同, 高于 PSO 算法, 在最大传输距离上与 PSO 算法相同, 达到了 112 km, 高于 LSA 算法, 且均能达到了相应算法的极限. PSLSA 算法的成本小于 PSO 算法和 LSA 算法之和, 因为 PSO 算法得到的初始值在一定程度上减少了 LSA 算法的迭代次数, 此外仅用 PSO 算法寻找非零点, 通过减少迭代次数, 可以有效地控制 PSO 算法的成本, 综合来看, PSLSA 算法具有有效性和可行性.

5 结 论

本文利用蒙特卡罗方法证明了密钥生成率关于可选参数的函数为非凸函数, 但同时也进一步对 LSA 算法能得到全局最优解进行了仿真分析, 这进一步增强了在量子密钥分发中, 利用 LSA 算法优化可选参数的科学性.

同时, 本文对 LSA 算法的强初始值依赖性进行了仿真分析, 结果表明无效的初始值对 LSA 算法具有致命性影响, 且在系统衰减较大时, 随机初始值有效的概率极小, 难以通过简单地尝试法判断是否存在使密钥生成率不为 0 的参数.

最后, 通过与其他算法的优化结果比较可知, 本文提出的 PSLSA 算法在密钥生成率和最大传输距离上都能达到较好性能, 虽然在成本上有所牺牲, 但是从实际应用来看, 文中算法都无法达到实时性要求, 这种时间上的牺牲, 不会对算法的应用场景产生更多的限制.

参考文献

[1] Shor P W, Preskill J 2000 *Phys. Rev. Lett.* **85** 441

[2] Pirandola S, Andersen U L, Banchi L, Berta M, Bunandar D, Colbeck R, Englund D, Gehring T, Lupo C, Ottaviani C, Pereira J L, Razavi M, Shamsul Shaari J, Tomamichel M, Usenko V C, Vallone G, Villoresi P, Wallden P 2020 *Adv. Opt. Photonics* **12** 1012

[3] Ekert A K 1991 *Phys. Rev. Lett.* **67** 661

[4] Lo H K, Chau H F 1999 *Science* **283** 2050

[5] Xu F, Ma X, Zhang Q, Lo H K, Pan J W 2020 *Rev. Mod. Phys.* **92** 025002

[6] Gerhardt I, Liu Q, Lamas-Linares A, Skaar J, Kurtsiefer C, Makarov V 2011 *Nat. Commun.* **2** 349

[7] Jain N, Stiller B, Khan I, Elser D, Marquardt C, Leuchs G 2016 *Contemp. Phys.* **57** 366

[8] Lo H K, Ma X, Chen K 2005 *Phys. Rev. Lett.* **94** 230504

[9] Lo H K, Curty M, Qi B 2012 *Phys. Rev. Lett.* **108** 130503

[10] Gu J, Cao X Y, Fu Y, He Z W, Yin Z J, Yin H L, Chen Z B 2022 *Sci. Bull.* **67** 2167

[11] Yin H L, Cao W F, Fu Y, Tang Y L, Liu Y, Chen T Y, Chen Z B 2014 *Opt. Lett.* **39** 5451

[12] Lo H K, Chau H F, Ardehali M 2005 *J. Cryptol.* **18** 133

[13] Wei Z, Wang W, Zhang Z, Gao M, Ma Z, Ma X 2013 *Sci. Rep.* **3** 2453

[14] Xu F, Xu H, Lo H K 2014 *Phys. Rev. A* **89** 052333

[15] Wang W, Lo H K 2019 *Phys. Rev. A* **100** 062334

[16] Ding H J, Liu J Y, Zhang C M, Wang Q 2020 *Quantum Inf. Process.* **19** 60

[17] Dong Q, Huang G, Cui W, Jiao R 2022 *Quantum Inf. Process.* **21** 233

[18] Dong Q, Huang G, Cui W, Jiao R 2021 *Quantum Sci. Technol.* **7** 015014

[19] Lu F Y, Yin Z Q, Wang C, Cui C H, Teng J, Wang S, Chen W, Huang W, Xu B J, Guo G C, Han Z F 2019 *J. Opt. Soc. Am. B* **36** B92

[20] Boyd S P, Vandenberghe L 2004 *Convex Optimization* (New York: Cambridge University Press) p716

[21] Han B L, Zhao R, Luo Q S, Xu F, Zhao J H 2017 *J. Beijing Inst. Technol.* **37** 461 (in Chinese) [韩宝玲, 赵锐, 罗庆生, 徐峰, 赵嘉珩 2017 北京理工大学学报 **37** 461]

[22] Jiang L, Ye R Z, Liang C Y, Lu W X 2019 *Computer Engineering and Applications* **55** 130 (in Chinese) [蒋丽, 叶润舟, 梁昌勇, 陆文星 2019 计算机工程与应用 **55** 130]

[23] Lucamarini M, Yuan Z L, Dynes J F, Shields A J 2018 *Nature* **557** 400

[24] Ma X, Zeng P, Zhou H 2018 *Phys. Rev. X* **8** 031043

[25] Zhou L, Lin J, Jing Y, Yuan Z 2023 *Nat. Commun.* **14** 928

[26] Zeng P, Zhou H, Wu W, Ma X 2022 *Nat. Commun.* **13** 3903

[27] Xie Y M, Lu Y S, Weng C X, Cao X Y, Jia Z Y, Bao Y, Wang Y, Fu Y, Yin H L, Chen Z B 2022 *PRX Quantum.* **3** 020315

[28] Ma X, Fung C H F, Razavi M 2012 *Phys. Rev. A* **86** 052305

[29] Sun S H, Gao M, Li C Y, Liang L M 2013 *Phys. Rev. A* **87** 052329

[30] Ma X, Qi B, Zhao Y, Lo H K 2005 *Phys. Rev. A* **72** 012326

[31] Curty M, Xu F, Cui W, Lim C C W, Tamaki K, Lo H K 2014 *Nat. Commun.* **5** 3732

[32] Xu F, Curty M, Qi B, Lo H K 2013 *New J. Phys.* **15** 113007

[33] Yang X S 2021 *Nature-Inspired Optimization Algorithms* (London: Elsevier) pp111–113

[34] Gandomi A H, Yun G J, Yang X S, Talatahari S 2013 *Commun. Nonlinear Sci. Numer. Simul.* **18** 327

[35] Ursin R, Tiefenbacher F, Schmitt-Manderbach T, et al. 2007 *Nat. Phys.* **3** 481

Improved parameter optimization method for measurement device independent protocol

Zhou Jiang-Ping Zhou Yuan-Yuan † Zhou Xue-Jun

(*College of Electronic Engineering, Naval University of Engineering Wuhan* 430000, *China*)

(Received 12 February 2023; revised manuscript received 19 April 2023)

Abstract

The optimal selection of parameters in practical quantum key distribution can greatly improve the key generation rate and maximum transmission distance of the system. Owing to the high cost of global search algorithm, local search algorithm is widely used. However, there are two shortcomings in local search algorithm. One is that the solution obtained is not always the global optimal solution, and the other is that the effectiveness of the algorithm is greatly dependent on the choice of initial value. This paper uses the Monte Carlo method to prove whether the key generation rate function is convex, and also simulates and analyzes the projection of the key generation rate function on each dimension of the parameter, in contrast to the approach in previous article. In order to eliminate the effect of the initial value, this paper proposes the particle swarm local search optimization algorithm which integrates particle swarm optimization algorithm and local search algorithm. The first step is to use the particle swarm optimization to find a valid parameter which leads to nonzero key generation rate, and the second step is to adopt the parameter as the initial value of local search algorithm to derive the global optimal solution. Then, the two algorithms are used to conduct simulation and their results are compared. The results show that the key generation rate function is non-convex because it does not satisfy the definition of a convex function, however, since the key generation rate function has only one non-zero stagnation point, the LSA algorithm can still obtain the global optimal solution with an appropriate initial value. When the transmission distance is relatively long, the local search algorithm is invalid because it is difficult to obtain an effective initial value by random value method. The particle swarm optimization algorithm can overcome this shortcoming and improve the maximum transmission distance of the system at the cost of slightly increasing the complexity of the algorithm.

Keywords: quantum key distribution, particle swarm algorithm, measurement device independent protocol, optimization of parameters

PACS: 03.67.Dd, 03.67.Ac, 03.67.Hk, 42.50.Ex **DOI:** 10.7498/aps.72.20230179

† Corresponding author. E-mail: EPJZYY@aliyun.com

物理学报 Acta Physica Sinica
Chinese Physical Society
Institute of Physics, CAS

改进的测量设备无关协议参数优化方法

周江平　周媛媛　周学军

Improved parameter optimization method for measurement device independent protocol

Zhou Jiang-Ping　Zhou Yuan-Yuan　Zhou Xue-Jun

引用信息 Citation: Acta Physica Sinica, 72, 120303 (2023)　DOI: 10.7498/aps.72.20230179
在线阅读 View online: https://doi.org/10.7498/aps.72.20230179
当期内容 View table of contents: http://wulixb.iphy.ac.cn
