文章编号:1671-4598(2018)03-0103-04 DOI:10.16526/j.cnki.11—4762/tp.2018.03.027 中图分类号:TP302.1 文献标识码:A

程序属性的检测与程序属性的分类

朱俚治
(南京航空航天大学 信息中心,南京 210016)

摘要:人们熟知的程序有恶意程序和善意程序,然而被病毒感染的程序具有某种特殊性质,可以定义为第三种类别的程序,第三种类别的程序是由本文提出。为了对三种不同类别的程序进行分类,本文提出解决问题的思路是:首先采用恶意权值计算公式对程序行为权值进行计算,判断该程序是善意程序还是恶意程序。如果是某个善意程序的权值发生了变化,那么该善意程序被病毒感染的可能性很大,因此对三种不同属性的程序分析后,最后使用MMTD算法对这三种程序进行分类:该程序是恶意程序,善意程序还是部分恶意部分善意的程序。

关键词:MMTD;善意;权值;恶意;病毒

Detection of Program Attributes and Classification of Program Attributes

Zhu Lizhi
(Information center, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract: people familiar with the program has a malicious program and goodwill program, but infected by the virus program has some special properties, can be defined as the third categories of program, the third categories of program is put forward by this paper. To classify three different categories of program, this paper puts forward the thought to solve the problem are: first, the malicious weight calculation formula is adopted to the program weight value calculate, the program is goodwill or malicious programs. If it is a goodwill program in the changes of weight value, so the possibility of the goodwill program by the virus infection is very big, therefore after analyzing three different attributes of the program, finally using MMTD algorithm to classify the three programs: this program is malicious programs, goodwill program or malicious part well-intentioned.

Keywords: MMTD; goodwill; weight; malware; virus

0 引言

病毒、蠕虫和木马都是恶意程序,然而日常在系统中运行的各种程序都为合法程序,这些程序都是善意的。然而传染性、攻击性和破坏性是病毒和恶意程序所具有的特性。

在恶意程序权值计算公式中,可以计算每一种程序的权值。根据该公式可以知道当某个程序的权值为正值时,就可以判断该程序为恶意程序,当某个程序权值为0或负值时,则可以判断该程序是善意程序。但是根据如上的叙述,可以知道恶意权值计算公式只能对两种属性程序进行描述。

然而实际上用户主机的系统中存在3种属性程序,即恶意程序、善意程序和被恶意病毒感染的程序。但是在恶意程序权值公式的计算中,并不能对被病毒感染程序进行判断和描述,恶意权值计算公式只能对一种程序进行判断,该程序是否为恶意程序,并不能对恶意程序进行属性上的区分,从而判断该程序是病毒还是被病毒感染的程序。病毒和被病毒感染的程序都具有恶意性,尽管病毒和被病毒感染的程序行为属性具有相似性,但从属性上来看这两种程序具有一定区别,而在这一点上是恶意权值计算公式所不能计算和判断的。根据上述原因为了将恶意程序、被病毒感染程序和善意程序进行区分,因此在本文提出一种算法,将这3种不同属性的程序进行定性区分。

收稿日期:2017-03-13; **修回日期**:2017-08-18。

作者简介:朱俚治(1980-),男,宜兴人,工程师,主要从事计算机技术与信息安全方向的研究。

在文献[3]中,该文还提出了恶意代码检测方案,最终的恶意性权值计算结果展现的是恶意代码对系统的破坏性。根据该文献[3]的定义2,可以知道计算恶意性权值反映的是恶意代码主体执行后,对系统的影响程度。病毒,合法程序和被病毒感染的程序是3种不同的程序,这3种程序对系统的影响程序是有所不同的,合法程序的权值为0或为负值,而病毒与被病毒感染程序的权值都大于0。作为有恶意行为的程序有两种程序,即病毒和被病毒感染的程序,然而纯粹的病毒与被病毒感染程序,对系统的影响程度有所差别,因此这两种程序的恶意权值有所差别,权值之大小是不一致的。

1 MMTD算法技术简介

中介真值程度度量知识简介:

在三值逻辑出现之前,只有二值逻辑。然而在二值逻辑中只有0与1两个真值,二值逻辑中的0与1在人们描述事物之属性时,只有对与不对和是与不是等等,因此在二值逻辑中,事物之属性值只存在两种属性,即事物的矛盾对立面。但事实上并非所有的事物仅仅存在矛盾对立面,还有些事物的属性存在反对对立面。我们把存在对立面的事物,同时又存在对立面之中介过渡状态的事物属性将采用一种新的逻辑去描述它,我们将能够描述这一事物的逻辑称为三值逻辑,这种三值逻辑通常称为中介逻辑。

中介真值程度度量方法,将是一种中介思想的具体应用,该方法符合中介思想。根据文献[2-3]对中介真值程度度量有如下的描述:中介真值程度度量的方法将事物的属性划分为五

种状态：事物的两个对立面，对立面的中间过渡状态和事物超态对立面[2-3]。这里用符号表示为 $\sim P$，P 与 $\neg P$，超态 ^{+}p 与超态 $\neg^{+}p$。

根据文献[2-3]相关的定义和说明，现在用数轴将中介真值程度度量的方法描述表达如下：

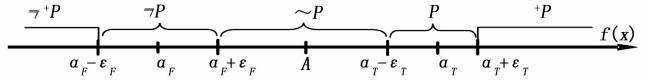

图 1 中介真值程度度量一维函数图

数轴上用符号 P 与 $\neg P$ 分别表示事物对立面的两个属性，符号 $\sim P$ 表示反对对立面之中间过渡状态事物的属性。

根据文献[1-2]可以知道在中介真值程度度量方法中，存在着两种距离比率函数（1）与（2），通过这两个距离比率函数，可以计算事物之属性值与 P，$\sim P$ 之间的距离。

1）计算事物属性值相对于 P 的距离比率函数[2-3]

$$h_T(x)=\begin{cases}\dfrac{-d(y,\alpha_F-\varepsilon_F)}{d(\alpha_T-\varepsilon_T,\alpha_F-\varepsilon_F)} & y<\alpha_F-\varepsilon_F\\ 0 & \alpha_F-\varepsilon_F<y<\alpha_F+\varepsilon_F\\ \dfrac{d(y,\alpha_F+\varepsilon_F)}{d(\alpha_T-\varepsilon_T,\alpha_F+\varepsilon_F)} & \alpha_F+\varepsilon_F<y<\alpha_T-\varepsilon_T\\ 1 & \alpha_T-\varepsilon_T<y<\alpha_T+\varepsilon_T\\ \dfrac{d(y,\alpha_F+\varepsilon_F)}{d(\alpha_T+\varepsilon_T,\alpha_F+\varepsilon_F)} & y>\alpha_T-\varepsilon_T\end{cases}$$

如果数轴上数值点的位置逐步接近于 P，则事物 A 具有 P 的属性逐步增强。

2）计算事物属性值相对于 $\neg P$ 的距离比率函数[2-3]

$$h_T(x)=\begin{cases}\dfrac{-d(y,\alpha_T+\varepsilon_T)}{d(\alpha_T+\varepsilon_T,\alpha_F+\varepsilon_F)} & y<\alpha_T+\varepsilon_T\\ 0 & \alpha_T-\varepsilon_T<y<\alpha_T+\varepsilon_T\\ \dfrac{d(y,\alpha_T+\varepsilon_T)}{d(\alpha_T-\varepsilon_T,\alpha_F+\varepsilon_F)} & \alpha_F+\varepsilon_F<y<\alpha_T-\varepsilon_T\\ 1 & \alpha_F-\varepsilon_F<y<\alpha_F+\varepsilon_F\\ \dfrac{d(y,\alpha_T-\varepsilon_T)}{d(\alpha_T-\varepsilon_T,\alpha_F-\varepsilon_F)} & y>\alpha_F-\varepsilon_F\end{cases}$$

如果数轴上数值点的位置逐步接近于 $\neg P$ 的取值区间，则事物 A 所具有的 $\neg P$ 属性逐步增强。

⑶如果该数值点的位置落在真值 $\neg P$ 和 P 的取范围之内，则事物 A 的属性部分地具有 $\neg P$ 的属性，同时又部分地具有 P 的属性。

2 MMTD 算法在检测恶意程序上的应用

2.1 恶意程序的定义

善意程序就是一种无害的程序，所谓的无害程序事实上就是为不会对其它程序造成破坏性的修改，对计算机资源使用时，不会造成破坏性的消耗，另外黑客不能通过该程序非法获取某些个人的敏感信息，这些程序都可以称为无害程序，无害程序也就是我们通常所说的合法程序，本文中将该程序称为善意程序。

恶意程序也可以称为有害程序，恶意程序的有害性，体现在对善意程序能进行破坏性的修改，或者成为黑客攻击和破坏计算机用户的系统和网络的某种工具，例如这些工具可以是蠕虫或木马，这些程序都可以称为有害程序，在本文中称为恶意程序。

2.2 恶意程序行为权值的计算

对于恶意代码来说，具有如下几种属性：传播性、激活性、保护性和破坏性，因此根据文献[1]中提出的公式中，分别将这几种行为属性作了如下的定义：$k_i(v)$、$k_e(v)$、$k_p(v)$、$k_d(v)$，分别定义为恶意代码的自传播性、自激活性、自保护性和自破坏性。

根据文献[1]可以知道，根据可疑程序的各个可疑指标，可计算可疑程序中的每一个行为属性，并且把某个程序不同的行为属性指数进行相加，就可以得出相应的结论，即该程序是恶意程序还是善意程序。恶意程序权值的计算公式如下：

根据文献[1]可以知道如下定义 2。

定义 2：独立的恶意权值计算，假设以 v 为主体行为共存在 n 个客体，每个客体系统的影响指数为 $k(o)=\alpha k_i(o)+\beta k_i(o)+\chi k_p(o)+\varepsilon k_d(o)$，那么主体恶意权值的计算公式为：$y_2=k(v)=\alpha k_i(v)+\beta k_i(v)+\chi k_p(v)+\varepsilon k_d(v)+\sum_{i=1}^{n}k(o_i)$[1]。

在恶意性权值计算公式中，符号 $k_i(v)$、$k_e(v)$、$k_p(v)$、$k_d(v)$ 分别表示客体，而每个恶意程序所具有的自传播性、自激活性、自保护性和破坏性可以分别称为某个恶意程序的客体，而每个恶意程序可以理解为某个主体。

在文献[1]中提出的计算公式，可以知道无论是恶意程序还是善意程序的某个客体值越大，则该程序行为属性对系统影响的程度就越大，如果客体的值为 0，则与该客体相对应的行为属性对系统没有产生任何影响。恶意代码对系统的影响性，主要是指恶意代码的各种行为属性对系统的影响，因此根据以上的叙述可以认为当某个主体的客体对系统没有产生影响时，客体的属性值则为 0。α、β、χ、δ 分别表示恶意代码行为：$k_i(v)$、$k_e(v)$、$k_p(v)$、$k_d(v)$ 的权重值。

根据文献[1]中的恶意程序权值计算公式有如下结论：如果某个程序是恶意程序，那么该程序的权值为正值，但如果某个程序是善意程序，那么该程序的权值就为 0 或负值。

2.3 恶意程序和合法程序权值的计算

病毒的传染性、激活性、破坏性和隐蔽性，都是病毒所具有的特征，然而正是由于这些行为的存在，所以通过恶意代码权值计算公式，所计算出来的程序行为属性的权值都不为 0，并且这些权值都是正值。

由于一般合法性程序不具有病毒的传染性、激活性、破坏性和隐蔽性，因此当采用恶意代码权值公式进行计算的时候，这些行为属性的权值都为 0。事实上在合法程序中某些程序具有了病毒类似的行为，例如病毒的的传染性、激活性、破坏性和隐蔽性等行为属性，但此时程序权值的计算结果为负值，因此这类属性的程序与病毒是两种不同类别的程序，从程序权值计算的角度可以将这两类程序相互区分。

1）根据以上的讨论有如下结论。

根据以上的分析可以知道，当程序的权值为正值时，则该程序就为恶意程序。由于病毒也是一种恶意程序，因此病毒的权值也为正值，但当得出的权值为 0 或负值时，该程序就为善意程序，在这里就是通常所说的合法程序。

2）被病毒感染的程序。

如果合法程序感染了病毒，那么该程序就具有了病毒的特

征。从程序权值的角度来看，此时程序的权值由 0 变成了正值，那么该程序的属性就发生了变化，具有了病毒的属性，因而此时该程序被病毒感染的可能性就很大。

如果某种病毒寄生于某个程序之后，当病毒没有发作之时，该程序表现的行为是正常的。但当病毒处于发作期的时候，那么该程序就具有了部分恶意性和攻击性，因此根据如上的叙述，就可以知道被病毒感染的程序具有了善意性，同时也具有了恶意性。

当程序体中的病毒处于发作期的时候，病毒和被病毒感染的程序都具有恶意性和攻击性，因此从恶意性的角度来看，这两种程序具有相似的行为特征。然而事实上合法程序被病毒寄生之前不具备恶意性。合法程序之所以具有恶意性和攻击性，只是因为此程序体中的病毒具有了感染性和破坏性，但由于被病毒感染的程序与病毒已融为一体，因此从程序行为属性上来看，被病毒寄生的程序确实具有了恶意性和攻击性。

尽管被病毒感染的程序具有某种程度上的恶意性，但并不是正真的病毒，这与纯粹的病毒是有的区别，因此从程序属性的角度来看，被病毒感染的程序是恶意程序和善意程序之间的过渡状态的程序，该程序部分具有恶意性同时又部分具有善意性。然而病毒是恶意代码，本身就具有攻击性和破坏性。根据如上的分析和叙述可以知道，病毒和被病毒感染的程序是有所区别的。

3　程序权值的计算和比较

$y = f(x)$ = 某种程序当前的权值 − 某种程序原始的权值

1）恶意程序权值的讨论：

如果某个程序的权值为正值时，则该程序就为恶意代码。只要该恶意代码的属性没有发生变化，那么权值就不会发生变化。病毒属于恶意代码，因此病毒的权值也始终为正值。根据上述原因有如下的结论：

当某种病毒的当前权值 = 某种病毒的原始权值 > 0 时，

则有 $y = f(x)$ = 某种病毒之当前权值 − 某种病毒的原始权值 = 0。

2）善意程序权值的讨论：

①如果某个程序的原始权值为 0，那么该程序则为善意程序。当程序的权值没有发生变化，则此时程序行为属性就没有发生变化，仍为善意程序，因此根据可以推断此时程序没有被病毒寄生和感染。

当某种合法程序的当前权值 = 某种合法程序的原始权值 = 0 时，则有 $y = f(x)$ = 某种合法程序之当前权值 − 某种合法程序的原始权值 = 0。

②如果某个程序的原始权值为 0，那么则该程序为善意程序。当程序的权值发生了变化，由 0 变成了正值时，那么此时程序的行为属性就发生了变化，具有了恶意性，因此可以推断此时的合法程序很可能被病毒感染了。

当某种合法程序的当前权值 > 0 时，而某种合法程序的原始权值 = 0 时，则有 $y = f(x)$ = 某种合法程序的当前权值 − 某种合法程序的原始权值 > 0。

③如果某个程序的原始权值为负值，而当前的权值没有发生变化，仍然为负值时，那么此程序的行为属性就没有发生变化，仍然为善意程序，因此可以推断此时的合法程序没有被病毒感染寄生。

当某种合法程序的当前权值 = 某种合法程序的原始权值 < 0 时，则有 $y = f(x)$ = 某种合法程序的当前权值 − 某种合法程序的原始权值 = 0。

④如果某个程序的原始权值为负值，而当前的权值发生了变化，由目前的负值变成了正值，那么此时程序的属性就发生了变化，由善意程序变成了恶意程序，因此可以推断此时的合法程序很可能被病毒感染寄生了。

当某种合法程序的当前权值 > 0，但是某种合法程序的原始权值 < 0 时，则有 $y = f(x)$ = 某种合法程序的当前权值 − 某种合法程序的原始权值 > 0。

4　MMTD 算法在程序属性判断方面的应用

4.1　程序权值比较公式

$y = f(x)$ = 某种程序当前的权值 − 某种程序的原始权值

根据以上的分析和讨论有如下结论：

1）采用恶意代码权值计算公式对某个程序进行计算。

①如果程序的权值为正值，则该程序就是恶意程序。

②如果程序的权值为 0 或负值，则该程序就是善意程序。

2）如果是善意程序，则有如下结论：

①如果善意程序的原始权值为 0 时，而当前的权值变成正值时，那么该程序就感染上了病毒。

②如果善意程序的原始权值为负值时，而当前的权值变为正值时，那么该程序同样感染上了病毒。

4.2　中介对恶意程序权值在匹配上的描述

1）根据程序的属性对程序进行分类：

恶意程序与善意程序是属性相反的两种程序，在中介逻辑中，将这两种程序的属性作为一个反对对立面。本文中所指的善意程序就是合法程序，恶意程序就是病毒。恶意和善意是两个反对对立面，而被病毒感染的程序部分具有恶意性，同时又部分具有善意性，因此当程序被病毒感染后，该程序所具有的属性应该处于中介逻辑所指出的中介过渡状态属性之处。

①当程序的权值为正值时，则为恶意程序，在中介逻辑中用符号 $\neg P$ 来表示，

其真值为 0。

②当程序的权值为 0 时或为负值，则为善意程序，在中介逻辑中用符号 P 来表示，

其真值为 1。

③当合法程序的权值发生了变化，则由原来的 0 或负值变为正值时，该程序就被病毒感染了，在中介逻辑中用符号表示 $\sim P$。

现用中介逻辑作如下分析和说明：数轴 $y = f(x)$ 上有 P，$\sim P$，$\neg P$ 三个数据的区域，P 代表善意程序，$\neg P$ 代表恶意程序，$\sim P$ 表示被病毒感染的程序。

从数轴上的 $y = f(x)$ 可以知道，在数轴上以 $\sim P$ 为对称中心，左右分别为 $\neg P$ 和 P。

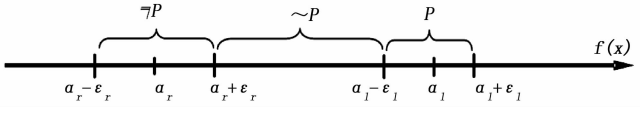

图 2　中介真值程度度量一维函数之应用

$y = f(x)$ 的值落在 3 个值域范围（$\alpha_r + \varepsilon_r, \alpha_l - \varepsilon_l$），（$\alpha_r - \varepsilon_r, \alpha_r + \varepsilon_r$）（$\alpha_l - \varepsilon_l, \alpha_l + \varepsilon_l$）之内。$\sim P$ 的区域为（$\alpha_r + \varepsilon_r$，

$\alpha_l-\varepsilon_l$），$\neg P$ 的区域为（$\alpha_r-\varepsilon_r,\alpha_r+\varepsilon_r$），$P$ 的区域为（$\alpha_l-\varepsilon_l$，$\alpha_l+\varepsilon_l$）。P 的真值为 1，$\neg P$ 的真值为 0。

①相对于 P 的距离比率函数

$$h_T(x)=\begin{cases}0 & \alpha_r-\varepsilon_r<y<\alpha_r+\varepsilon_r\\ \dfrac{d(y,\alpha_r-\varepsilon_r)}{d(\alpha_l-\varepsilon_l,\alpha_r+\varepsilon_r)} & \alpha_r+\varepsilon_r<y<\alpha_l-\varepsilon_l\\ 1 & \alpha_l-\varepsilon_l<y<\alpha\alpha_l+\varepsilon_l\end{cases}$$

通过距离比率函数 $h_T(x)$ 对 y 值的计算有如下结论，

①函数 $h_T(x)=1$，y 之值落在区域（$\alpha_l-\varepsilon_l,\alpha_l+\varepsilon_l$）中，则为善意程序。

②若函数 $h_T(x)=0$，y 之值落在区域（$\alpha_r-\varepsilon_r,\alpha_r+\varepsilon_r$）中，则为恶意程序。

③若函数 $h_T(x)=\dfrac{d(y,\alpha_r-\varepsilon_r)}{d(\alpha_l-\varepsilon_l,\alpha_r+\varepsilon_r)}$，y 的值落在区域（$\alpha_r+\varepsilon_r,\alpha_l-\varepsilon_l$）中，则为被病毒感染的程序。

2）本文提出的算法分析：

①具有恶意行为的程序包含两种情况：1. 纯粹的病毒；2. 被病毒感染的程序。

当被病毒寄生的程序体中的病毒没有发作的时候，此时程序的行为是善意的。

当被病毒寄生的程序体中的病毒开始发作的时候，此时程序的行为就体现出一定的恶意性。

如果原始程序本身就是病毒，程序的权值则始终为正值。

②在善意程序中，提取原始程序的权值与当前程序的权值进行比较。

如果原来是善意程序，则程序的权值没有发生变化，那么此时仍然是善意程序。

如果程序的权值发生了变化，则由原来的 0 或者负值变为正值时，那么此时该程序就被病毒感染了。

由于在被病毒感染的程序中，程序体中的病毒没有发作时，该程序体现出的仍然是善意性。

由于被病毒感染的程序中，如果程序体中的病毒处于发作状态时，那么该程序就表现出恶意性。

如果原始程序本身是善意程序，但经过病毒的感染，使得被寄生的程序具有了恶意性，那么此时程序的权值就发生了变化。由原来的 0 或者由负值变成了正值。因此当程序的权值发生变化后，那么该程序就很可能被病毒感染了，该善意程序具有了一定的恶意性。

5 结束语

在通过恶意权值计算公式对程序进行计算时，可以判断该程序是善意的还是恶意的。通过程序权值计算还可以发现，某个程序的权值是否发生了变化。如果某个合法程序的权值发生了变化，则该程就被病毒感染了，被病毒感染的程序，就体现出部分恶意性和部分善意性。因此在本文中首先通过恶意权值计算公式对某个程序的属性进行判断，然后再使用 MMTD 算法对 3 种属性的程序进行分类，由本文提出的算法就达到了对病毒检测之目的，同时也实现了对不同属性程序分类之目的[4-11]。

参考文献：

[1] 刘巍伟，石 勇，郭 韩，等. 一种基于综合行为特征的恶意代码识别方 [J]. 电子学报，2009 (9)：696－700.
[2] 洪 龙，肖奚安，朱梧槚. 中介真值程度的度量及其应用 (I) [J]. 计算机学报. 2006 (12)：2186－2193.
[3] 朱梧槚，肖奚安. 数学基础与模糊数学基础 [J]. 自然杂志. 7 (1980)：723－726.
[4] 张波云，殷建平，张鼎兴，等. 基于集成神经网络的计算机病毒检测方法 [J]. 计算机工程与应用，2007，43 (13)：26－29.
[5] 贺朝晖. 计算机病毒对抗检测高级技术分析 [J]. 计算机安全，2010 (10)：93－97.
[6] 张海波. 计算机病毒检测技术研究 [J]. 计算机光盘与应用，2014 (13)：181－182.
[7] 张岳公，范希骏，李大兴. 计算机病毒入侵检测技术探讨 [J]. 微电子学与计算机，2005，22 (11)：33－38.
[8] 张森强，郭兴阳，唐朝京. 检测多态计算机病毒的数学模型 [J]. 计算机工程，2004，30 (17)：24－25.
[9] 陈 桓，刘晓洁，宋程粱. 一种基于免疫的计算机病毒检测方法 [J]. 计算机应用研究，2005 (9)：111－114.
[10] 刘 俭，唐朝京，张森强. 一种计算机病毒的检测方法 [J]. 计算机工程，2004，30 (6)：127－129.
[11] 邢小东，侯 飞，李千路. 一种应用免疫原理的计算机病毒检测方法研究 [J]. 计算机安全，2011 (2)：39－42.

(上接第 93 页)

响巨大，设计出一种作战用固定翼无人机落地姿态平衡控制系统，对系统硬件部分的主要电路进行介绍，优化设计系统软件部分，采用磁强计和加速度计对作战用固定翼无人机落地姿态进行测量，通过实验验证了该系统的性能，为以后固定翼无人机飞控系统的研究奠定了基础。

参考文献：

[1] 薛 亮，王新华，贾 森，等. 基于模糊 PID 的多旋翼无人机姿态控制系统设计 [J]. 电子设计工程，2016，24 (16)：61－63.
[2] 张兴文，陈 铭，曹 飞，等. 无人机姿态控制系统设计及仿真 [J]. 计算机仿真，2016，33 (7)：158－161.
[3] 刘 强. 四旋翼飞行器控制系统的研究 [J]. 科技通报，2016，32 (4)：121－125.
[4] 傅忠云，朱海霞，孙金秋，等. 两轮载人自平衡车姿态测量单元设计 [J]. 科学技术与工程，2015，15 (15)：66－71.
[5] 张 鹏，王 键. 小型固定翼无人机纵向姿态控制律的研究 [J]. 计算机测量与控制，2015，23 (8)：2686－2688.
[6] 费爱玲，李 柠，李少远. 固定翼无人机的自抗扰反步控制 [J]. 控制理论与应用，2016，33 (10)：1296－1302.
[7] 董守田，杨利红，康成吉，等. 固定翼无人机姿态控制及仿真 [J]. 东北农业大学学报，2015，46 (9)：87－92.
[8] 曹美会，鲜 斌，张 旭，等. 基于视觉的四旋翼无人机自主定位与控制系统 [J]. 信息与控制，2015，44 (2)：190－196.
[9] 赵海盟，张文凯，谷静博，等. 无人机载荷航拍控制系统设计 [J]. 计算机应用，2015，35 (1)：270－275.
[10] 江 杰，冯旭光，苏建彬. 四旋翼无人机仿真控制系统设计 [J]. 电光与控制，2015，22 (2)：27－30.