2019-12

OPTIMIZING TRAINING EVENT SCHEDULES AT NAVAL AIR STATION KINGSVILLE

Meditz, Malia

Monterey, CA; Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

OPTIMIZING TRAINING EVENT SCHEDULES AT NAVAL AIR STATION KINGSVILLE

by

Malia Meditz

December 2019

Thesis Advisor: Robert F. Dell
Second Reader: Jeffrey F. Hyink

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** December 2019	**3. REPORT TYPE AND DATES COVERED** Master's thesis		
4. TITLE AND SUBTITLE OPTIMIZING TRAINING EVENT SCHEDULES AT NAVAL AIR STATION KINGSVILLE				**5. FUNDING NUMBERS**
6. AUTHOR(S) Malia Meditz				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000				**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.				**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

VT-22 located at Naval Air Station Kingsville is a U.S. Navy and Marine Corps Strike and E-2/C-2 training squadron. At VT-22, naval instructor pilots with fleet experience train student aviators through a substantial and challenging syllabus. Schedulers at VT-22 create each day's schedule manually using a laborious process that leaves little time to explore options or consider a time horizon beyond one day. This thesis develops, implements, and reports on the Training Event Scheduling Tool (TEST), an integer linear program that prescribes hourly assignments of classroom events, simulator events, and flight events to each student and each instructor. TEST streamlines scheduling using optimization to simultaneously create daily schedules for a horizon of up to one week. We compare TEST schedules with those created manually for a typical week and find that TEST schedules up to 32% more events. TEST solution time is less than 10 minutes for a single day's schedule, allowing the heretofore unavailable opportunity to quickly explore schedule options and respond to changing requirements.

14. SUBJECT TERMS flight schedules, scheduling, schedule optimization, aviator training, flight training, pilot training, advanced strike training, intermediate strike training, naval aviators, Marine Corps aviators, pilots			**15. NUMBER OF PAGES** 69
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

OPTIMIZING TRAINING EVENT SCHEDULES AT NAVAL AIR STATION KINGSVILLE

Malia Meditz
Ensign, United States Navy
BAS, U.S. Naval Academy, 2018

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 2019

Approved by: Robert F. Dell
Advisor

Jeffrey F. Hyink
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

VT-22, located at Naval Air Station Kingsville, is a U.S. Navy and Marine Corps Strike and E-2/C-2 training squadron. At VT-22, naval instructor pilots with fleet experience train student aviators through a substantial and challenging syllabus. Schedulers at VT-22 create each day's schedule manually using a laborious process that leaves little time to explore options or consider a time horizon beyond one day. This thesis develops, implements, and reports on the Training Event Scheduling Tool (TEST), an integer linear program that prescribes hourly assignments of classroom events, simulator events, and flight events to each student and each instructor. TEST streamlines scheduling using optimization to simultaneously create daily schedules for a horizon of up to one week. We compare TEST schedules with those created manually for a typical week and find that TEST schedules up to 32% more events. TEST solution time is less than 10 minutes for a single day's schedule, allowing the heretofore unavailable opportunity to quickly explore schedule options and respond to changing requirements.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

C2	C-2 Greyhound (operated by U.S. Navy)
CQ	carrier qualification
E2	E-2 Hawkeye (operated by U.S. Navy)
FIST	Flight Instructor Standardization and Training
FRS	Fleet Replacement Squadron
FTS	Flight Training Scheduler
LSO	landing signal officer
NAS	naval air station
SAT	Scheduling Assistance Tool
SKEDSO	scheduling officer
SNA	student naval aviator
T-45	McDonnell Douglas T-45 Goshawk (operated by U.S. Navy)
TCM	Training Capability Model
TEST	Training Event Scheduling Tool
TIMS	Training Integration Management System
TW-2	Training Air Wing Two
VFA-106	Strike Fighter Squadron 106
VT-22	Training Squadron Twenty Two

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The United States Navy trains its own pilots through an extensive pipeline that challenges student aviators physically and mentally. The Navy assigns an aircraft platform to each student based on the Navy's resource requirements, the student's overall score in flight school, and the student's preferences. Students in Intermediate and Advanced Strike training go to either Naval Air Station (NAS) Kingsville, Texas or NAS Meridian, Mississippi. These students undergo an extremely rigorous and complex syllabus that includes flight events, simulator events, and lecture events. At NAS Kingsville, experienced instructor pilots who have flown naval aircraft in the fleet train more than 100 students at a time. Student aviators who complete their Advanced Strike training phase, receive "Wings of Gold" as Naval Aviators and are qualified to move onward to operate some of the most innovative and advanced technology in the military.

There is currently a shortage of qualified pilots in the Navy, which is largely due to the backlog occurring in the flight training pipeline. One way to decrease all student aviators' time-to-train is by creating efficient daily schedules that meet the goals of training squadrons and maximize the number of required events. Creating an efficient daily schedule is no easy task because it must include a variety of constraints, limitations, and resources. Training Squadron Twenty Two (VT-22), a jet training squadron located at NAS Kingsville, identified a need to optimize daily schedules. Currently, schedulers manually create daily schedules for about 120 students and 30 instructors, which they post daily online at 1600. At about 80 events per day, manually generating a schedule is a laborious task that leaves little time to explore options.

In this thesis, we address VT-22's scheduling goals by developing, implementing, and testing an integer linear program called Training Event Scheduling Tool (TEST) that prescribes hourly assignments of classroom events, simulator events, and flight events to each student and each instructor at NAS Kingsville. TEST streamlines scheduling using optimization to simultaneously create daily schedules for a horizon of up to one week. Schedulers can run TEST to produce either daily or weekly schedules with instructor and

student pairings that adhere to the syllabus requirements of Intermediate training and Advanced training.

TEST reads data from Excel spreadsheets and outputs a spreadsheet containing all names of students and instructors, event assignments, and periods for the time horizon specified (daily or weekly). For each student, TEST takes as input: (a) “desired” events that the schedulers want the student to complete, (b) completed events, (c) periods of non-availability, and (d) the stage or stages of training to be considered. A student’s training stage (or stages) provides a list of lower priority events for scheduling that we call the student’s “possible” event list. For each instructor, TEST takes as input: (a) qualifications and (b) periods of non-availability. TEST also includes other inputs such as prerequisite events, event length, number of instructors needed for each event, and jet availability.

TEST accommodates VT-22’s scheduling goals and enables scheduling more events in less time compared to schedules that are created manually. We compare TEST using real data collected by VT-22 over one week. We consider two cases: good weather and bad weather (accounting for real-world conditions). For bad weather, we do not allow TEST to schedule any events in periods when weather restricted flight events from occurring. It could be argued that TEST has an unfair advantage due to the fact that it has perfect weather information and knows exactly which periods need to be blocked out. While this may slightly skew results, it shows the value of using weather forecasts to predict which periods may become unavailable for flying so that simulator events and lecture events can be more effectively planned. With bad weather, TEST schedules a total of 21% more events compared to VT-22’s current schedules. The 21% total improvement is achieved with a 46% increase to the number of lecture events, a 59% increase to the number of simulator events, and slightly less (about 4%) flight events. When we assume good weather, TEST schedules 32% more events and is better in all three event categories.

Current daily schedules require one full workday to manually create. TEST generates a daily schedule in less than 10 minutes and a weekly schedule in a few hours. VT-22 schedulers can run TEST over the weekend to output a weekly schedule that forecasts the week ahead. As flights may be canceled or incomplete throughout the week due to weather, bird activity, jet maintenance, or other reasons, schedulers can run a daily instance of TEST to ensure that students progress quickly through the syllabus.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank my thesis advisor Dr. Robert Dell for all of his expertise and guidance throughout my entire time at NPS. He taught me much more than just writing a good integer linear program. I would also like to thank everyone who helped contribute to this thesis effort including the VT-22 schedulers and officers and my second reader CAPT Jeffrey Hyink.

Furthermore, I would like to thank all of my family and friends who supported me throughout this time. To my mom, thank you for putting up with my daily phone calls of random musings and rantings. To my dad, thank you for always being just a phone call away whenever any unavoidable issues crossed into my lane. To Alana and Justin, thank you for constantly making me laugh even when I was under a lot of stress. I would also like to thank my NPS friends who forced me to take breaks, have fun, and just enjoy beautiful Monterey every once in a while.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The United States Navy trains its own pilots through an extensive pipeline. Flight school equips Navy pilots to operate some of the most innovative and advanced technology in the military. Each Student Naval Aviator (SNA) is challenged physically and mentally in numerous classes including engineering, aerodynamics, and water survival throughout his or her time in flight school (U.S. Navy 2018). SNAs begin flight school training in Pensacola, Florida, with the Aviation Preflight Indoctrination. Once completed, they move on to Primary. At the end of Primary training, the Navy assigns a platform to each SNA based on the Navy's resource requirements, the student's overall score in flight school, and the student's preferences. Intermediate and Advanced phase training at Naval Air Station (NAS) Kingsville or NAS Meridian is the next step for a SNA meeting a minimum score and selected to fly jets. The Strike pipeline includes an extremely rigorous syllabus of flight events, simulator events, and classroom events requiring around a year to complete. NAS Kingsville trains over 100 students at a time with naval instructor pilots who have experience flying naval aircraft in the fleet. Students conduct flight events in the McDonnell Douglas T-45 *Goshawk* (T-45) aircraft. Upon completion of Advanced, the student aviators receive their "Wings of Gold" and report to a Fleet Replacement Squadron (FRS).

The Training Event Scheduling Tool (TEST), a result of this thesis, is an integer linear program that prescribes hourly assignments of classroom events, simulator events, and flight events to each student and instructor at NAS Kingsville. TEST streamlines scheduling using optimization to create efficient daily schedules quickly. Correspondence with NAS Kingsville identified the need for a tool that not only outputs optimized schedules; but, one that also accommodates all of the varying resources and constraints (Goodwin 2018). TEST accomplishes this by scheduling more events in less time compared to the manually created schedules. TEST outputs schedules up to seven days with instructor and student pairings that adhere to the Intermediate and Advanced syllabus requirements.

A. BACKGROUND

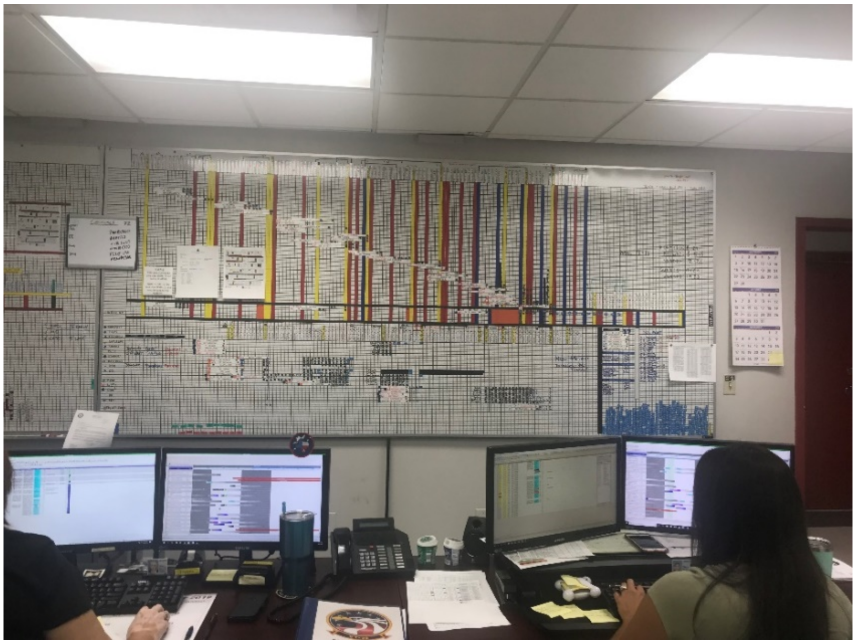
Training Air Wing Two (TW-2), located at NAS Kingsville (Figure 1), trains and graduates over 200 students per year as qualified U.S. Navy and Marine Corps jet pilots. Within TW-2 are Training Squadron Twenty One and Training Squadron Twenty Two (VT-22). VT-22 Golden Eagles instructor pilots train SNAs who select either jets or the E-2 Hawkeye and C-2 Greyhound (E2/C2) platform. Students who select E2/C2, complete the Intermediate phase before heading to VT-22 for Advanced training. E2/C2 Advanced training takes just over 34 weeks to complete (Naval Air Training Command 2014a). The syllabus for the E2/C2 track requires 60 simulator events, 75 flights in the T-45, and 158 lecture hours. For the Strike pipeline, students begin their training in the Intermediate jet course lasting approximately 27 weeks. Upon completion of Intermediate training, students transition into the Advanced Strike course lasting approximately 25 weeks for TW-2 (Naval Air Training Command 2014b). At the end of 52 weeks, students complete at least 73 simulator events, 133 flights in the T-45 and 205 lecture hours to earn their "Wings of Gold" and continue on to an FRS as qualified Strike pilots.

Figure 1. VT-22 at NAS Kingsville in front of the T-45 *Goshawk* (VT-22 Golden Eagles)

B. SCHEDULING TRAINING EVENTS

Due to the high volume of student pilots and the complexity of training events required for the VT-22 syllabus, scheduling training is no easy task. Currently, scheduling officers (SKEDSOs), also called schedulers, create daily schedules manually, which can lead to an inefficient use of resources—students, instructors, and aircraft. At around 80 flight events per day, creating schedules that account for syllabus requirements, crew day limitations, non-availabilities, and other constraints is time consuming with little opportunity to explore options to improve schedules. Inefficient schedules can lead to playing catch-up on typical “off days.”

The Scheduling Office begins each day with the goal of creating the next day’s schedule by 1600. Schedulers create the schedule, post it for the squadron to see, go home, and the cycle continues the next day. The schedulers’ motto is to “write the schedule early, and write the schedule often.” They could potentially plan schedules days into the future. However, they seldom do so because of the time required and revision that oftentimes occurs due to unforeseen circumstances. Each day starts with an evaluation of the previous day to identify which events were not completed, whether it was due to a student inefficiency, weather, or any other disruption. The Scheduling Office uses a detailed whiteboard (Figure 2) with the entire syllabus flow, student names, instructor names, and magnetic markers representing student progress to assist with ensuring SKEDSOs schedule students for the right events.



Author photo, taken on September 26, 2019.

Figure 2. White board used and updated daily to create schedules

Every morning, the schedulers update the white board and begin assigning instructors to events and students (we use student and SNA interchangeably). Higher priority is given to those students who have to complete certain stages in time for hard deadlines such as Carrier Qualification (CQ) detachments. Students nearing a warmup window, a period of time in which students fall out of currency, are typically also given higher priority. SKEDSOs record the number of days since each student's last flight. For a student to maintain currency, the student must not exceed the maximum number of days allowed between flights. If a student enters a warmup window, or a lack of currency, he or she needs to complete a warmup flight event before continuing through the syllabus. Prioritizing SNAs nearing a warmup window can help decrease overall time to train. In the past, schedulers might have also given higher priority to students who are nearing the end of the syllabus. However, according to experienced schedulers at VT-22, this leads to a cyclic trend with underlying consequences for steady progress and output.

1. Syllabus

The syllabus for Strike training divides into two *phases*: Intermediate and Advanced. Intermediate, or Phase I, is the first that SNAs must complete. Once a student completes all events in Phase I he or she can move on to Advanced, or Phase II. Within each phase are *stages* such as Familiarization, Instrument Rating, and Carrier Qualifications. Within each stage are *blocks* such as Familiarization Solo, Instrument Rating Simulator, and Carrier Qualification Landing Flight Procedures. Many stages include a combination of lecture, exam, simulator, and flight events. The first events in a stage are typically lecture events taught by both military and civilian personnel. Simulator events, instructed by civilian personnel, usually come next. The last event in a stage is oftentimes a student solo with multiple flight events preceding this. Each event has a list of prerequisites that a student must complete. Phase I is mostly "linear" in that the prerequisites require each student to follow the same sequence of events with little variability. Thus, scheduling events for students in Phase I is usually consistent from one student to the next. However, Phase II (Figure 3) allows for more flexibility. A student may traverse multiple paths as long as he or she completes all the prerequisites for a specific event.

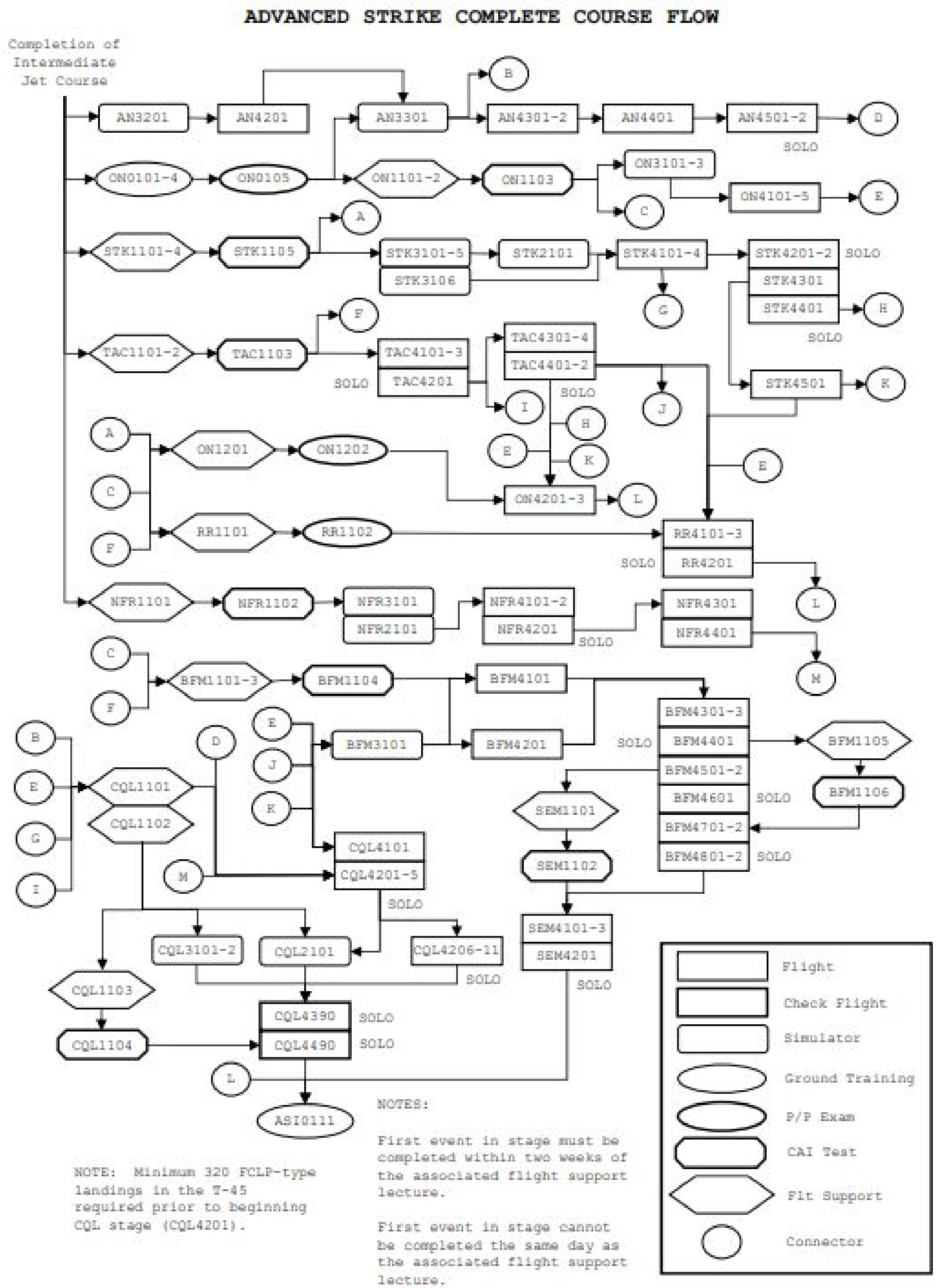


Figure 3. Advanced Strike complete course flow.
Source: Naval Air Training Command (2014b).

2. Instructors

Both military and civilian personnel instruct students. Civilian personnel teach a specific subset of all lecture events as well as all of the simulator events. Military instructor pilots (henceforth instructors) teach the remaining lecture events as well as all of the flight events. Each instructor has qualifications for a subset of all flight events. Schedulers use the Flight Instructor Standardization and Training (FIST) matrix (Figure 4) on a daily basis to identify instructor qualifications. SKEDSOs are responsible for ensuring that only qualified instructors teach students in specific events as well as allowing for some flexibility so instructors can remain current in existing qualifications and finish additional qualifications. Currently, the schedulers view instructors as the most limiting resource and therefore strive to schedule each instructor to two events per day.

Quals Complete Report

Unit: VT-22
Report Generated: 4//2019 7:46:23 AM
Classification: Instructor Pilot
Status: All
Personnel: Multiple
FITU Stan IPs
Reservist

Quals Level

I = IUT Q = Qualified
D = Designated X = Checker
F = Facilitator S = Stan checker
E = Expired L = Lead
C = Current

Q SORTIE Needed (black font)
Q EXAM Needed (white font)
E STAN CHK w/ "S" And EXAM Needed

Expiration > 60 days
Expiration 30 - 60 days
Expiration 15 - 29 days
Expiration < 14 days
Expired

Rank Full Name	INST	FAM	FRM	SECTLD	DIV	DIVLD	NFM	NFMCHASE	NFR	ON	RR	SLL	STK	STKLD	OCF	TAC	TACLD	BFM	BFMLD	SEM	SEMLD	CQDEMO	LDSF	LSO	FCF	INSTCK	NATOPS	CRM	T-45C IP
LT A, R	S	S	X	L	X	L	X	X	X						X	I										D		S	D
Maj B, D	X	X	X	L	X	L	X		X	X	I	I	X	D	I	X	D	I										X	D
LT B, J	X	X	X	L	X	L	X	X	X							I						X	I	I					D
LT C, A	S	S	S	L	S	L	S		S						E	S	D	S	D	S	D	X	S	D					D
Capt C, D	X	X	X	L	X	L	X		X	X			X			I	I												D
CDR C, C	S	S	S	L	S	L	S		S	X	I	X	X	D	X	X	D	I				X	L			D		X	D
LCDR C, N	S	S	S	L	S	L	S		S																	D		X	E
LCDR F, W	X	X	X	L	X	L	X		X						X	X	D	I				X	L	I					D
LtCol G, K	X	X	X	L	X	L	X		X	X	I	I	X	D	I	X	D	I											D
LT G, M	S	S	S	L	S	L	S		S	S	S	S	S	D	S	S	D	S	D	S	D	X	E				D	X	D
Maj H, B	S	X	X	L	X	L	X		X	X	X	X	X	D	X	X	D	X	D	Q						D		X	D
LT J, K	X	X	X	L	X	L	X		X						X	X	D	X	I			X	L	D	I				D
1stLt K, P	X	Q	X		X		X		Q																				D
LT K, J	S	S	S	L	S	L	S	X	S							Q						X	L	D		D		X	D
LT L, A	S	S	S	L	S	L	E		E	S	S	S	S	D	E	S	D	S	D	X	I				I		D	X	D
LT M, J	S	S	S	L	S	L	S		S	S	S	S	S	D	S	S	D	S	D	E									D
LCDR M, N	S	S	S	L	S	L	S		S	S	S	S	X	D		S	D									D	D	X	D
Maj N, M	X	X	X	L	X	I	X		X	I			I																D
LT P, B	X	X	X	L	X	L	X		X																				E
LT R, C	X	X	X	L	X		X		X	I			I																D
Maj S, D	X	X	X	L	X	I	X		X	I			I																D
LT S, S	Q	Q	Q		Q		X		Q																				D

Figure 4. Example Flight Instructor Standardization and Training Report. Adapted from Simpson (2019).

3. Current System

Currently, schedulers use the Training Integration Management System (TIMS) to record schedules (Naval Air Training Command 2017). TIMS identifies crew rest as well as non-availability periods. Unfortunately, it is an aged process; it does not produce an error for combinations of inputs that violate syllabus requirements. It is also tedious in that it separates lecture events, flight events, and simulator events into different tabs. Schedulers coordinate with simulator instructors, who are civilian personnel located in a separate building on the base, when it comes to assigning simulator events. The Scheduling Office at VT-22 has the hours of operation for the simulators but not specific instructor availabilities. Thus, before the SKEDSOs can post the final schedule, they have to go through a few iterations with the simulator unit.

4. Business Rules

The schedulers must give consideration to a variety of constraints to include crew rest, instructor qualifications, non-availabilities, night flights, and prerequisites. Before schedulers pair a student and instructor with an event, that student must have successfully completed all prior prerequisite events. In the case of the whiteboard, in general, a student should have already completed all events to the left of his or her marker. Often, schedulers must ensure a SNA completes all events in a stage before moving on to the next. Additionally, some flights require students conduct them as night flights, and students need to maintain a certain number of night hours throughout the syllabus.

Crew day and crew rest are huge considerations that schedulers must not violate. Every instructor and student should have a crew day that does not exceed 12 hours and crew rest no less than 12 hours (Naval Air Training Command 2014b). Additionally, student days must not exceed eight hours of lectures. If a student has a flight or simulator event on a given day, he or she can have up to four additional lecture hours. A student can have at most two flights or simulator events per day. A few events are so demanding that at most one be completed without any other events in the same day. VT-22 instructors have the same restrictions as students in terms of instructing lectures and flying events.

However, schedulers may occasionally "surge" the number of events an instructor teaches. Instead of just two flights, they may assign three events to an instructor per day.

There are a variety of other constraints including CQ specifications, jet availability, simulator capacities, specific event restrictions, on-wing instructors, and a minimum number of flights that schedulers must also consider. Events in the CQ stage are special because they require a landing signal officer (LSO) to wave students at the runway. LSOs can wave up to five students at a time at night and up to six students during the day. While in the CQ stage, SNAs can complete up to three flights per day, and they should not be in any other stage. Schedulers must also ensure there are enough jets available for each flight event. Similarly, they must ensure that there is enough capacity at the simulators for all of the simulator events. The simulators can accommodate up to five students at a given start time. Some events must be scheduled consecutively while others need to be completed on different days such as the last lecture and the first simulator or flight event in any stage. This ensures that students have enough time to digest the lecture material and prepare for their simulator and flight events. Schedulers need to not only ensure they assign qualified instructors to certain events but also adhere to on-wing requirements. An on-wing is "one of two primary instructors assigned to prepare a student in the Familiarization stage" (Naval Air Training Command 2014b, p. xxii). Other events require SNAs not to fly with their on-wing. Additionally, to keep students out of a warmup window by remaining current with flights, schedulers aim to assign students to a minimum number of flights per week. All of these aspects lend to the difficult nature of scheduling events in this syllabus.

5. Output

The Scheduling Office aims to deliver "good" schedules where "good" is not explicitly defined but is based on the experience of those who have been scheduling VT-22 events for over ten years. At the end of the day, they hope to create a robust schedule that can overcome disruptions when events do not go as planned such as last-minute cancellations due to weather or student failures. One way they achieve this is to explore how cancellations earlier in the day affect flights later in the day. Schedulers release the next day's schedule at 1600. This deadline gives students enough time to prepare for their flights the next day, it allows the schedulers to add any last-minute changes with information from earlier in the day, and it accounts for morale of the entire squadron. Schedulers output the daily schedules in an easy-to-read format that VT-22 personnel can sort by time, type of event, instructor, student, or length of event.

C. TEST

This thesis creates the Training Event Scheduling Tool, an optimization tool using integer linear programming that not only outputs up to one week's worth of schedules but also takes into consideration the main goal of VT-22: making winged aviators. TEST can aid the Scheduling Office with creating daily flight schedules that have minimal deviance from a student's expected timeline. TEST allows schedulers to visualize up to seven daily schedules as opposed to just the next day's manually generated schedule. Our tool considers many limitations and follows all of the syllabus requirements for both Intermediate and Advanced phases. TEST has the ability to be quickly rerun with new inputs in the case of incomplete events that happen earlier in the day.

II. LITERATURE REVIEW

There is a plethora of published research on scheduling problems both related to the military and the civilian sector. However, there is little work on flight scheduling in the military specifically. In this chapter, we highlight a few recent projects on the topic of scheduling in the military that are most closely related to the effort of this thesis. We also explore the idea of *persistence* as described by Brown, Dell, and Wood (1997), which we use to guide formulation of our TEST integer linear program.

Using optimization and integer linear programs to create efficient schedules is something that people implement on a daily basis for a variety of situations. Some common examples are class schedules and airline flights. One reoccurring issue with scheduling is a “lack of *persistence*” (Brown et al. 1997, p. 16). Because a scheduling problem depends on time, it makes sense that assignments scheduled later in the day are not always independent of those scheduled earlier. Thus, if one assignment does not go as planned according to the schedule, the remainder of the schedule may be unusable. In the case of airline flights, a flight canceled earlier in the day can greatly alter flights scheduled later in the day, which results in delays and more cancellations. One solution to this problem is to rerun the scheduling model with the updated inputs to produce a new schedule based on the new information. However, a model that is not rooted in the idea of *persistence* can output a schedule that is drastically different than the previous one with only minimally different inputs. This is not ideal for most businesses. Brown et al. (1997) write about the importance of creating a persistent model. There are a few ways to create a persistent model, and one of them is to convert “the original objective function into an aspiration constraint” so that “the objective function of the persistent model is a surrogate objective that just measures deviations from the original schedule” (Brown et al. 1997, p. 18). A persistent model would include an objective function that imposes penalties on creating a schedule that greatly differs from the desired schedule, which is the goal for scheduling VT-22 training events as well as many other cases.

Scheduling across the aviation communities is a similar process that is often done manually by SKEDSOs, but every squadron’s syllabus is different. Thus, it is difficult to extend previous work done on scheduling from one command to another due to varying resources, goals, and business rules. However, a former Naval Postgraduate School student, Roger Jacobs, studied this same scheduling problem concerning VT-22's flight schedules. He developed the Flight Training Scheduler (FTS), which outputs seven days' worth of schedules in 30 minutes using the General Algebraic Modeling System (Jacobs 2014). There are a few major features missing from FTS that deter the SKEDSOs at NAS Kingsville from using this model on a daily basis. First, FTS does not implement crew rest for instructors and students. Instructors and students need at least 12 hours to rest beginning after their last scheduled event on one day and ending at the start of their next event on the following day (Naval Air Training Command 2014b). FTS "does not coordinate periods to time," and thus does not have the capability of implementing crew rest (Jacobs 2014, p. 35). Second, FTS only accommodates six periods in each day, but this does not give SKEDSOs the necessary resolution they normally incorporate into their manual schedules. Periods that are an hour or shorter in length would provide a much more realistic schedule with less manual intervention. Third, Jacobs only considered scheduling flight events in the Advanced Strike syllabus. This can cause problems for the schedulers because students in the Intermediate phase share the same resources as those in the Advanced phase. Using FTS, instructors that are assigned to students in the Advanced phase would need to be considered unavailable to instruct those in the Intermediate phase. Fourth, lectures, night events, and simulator events are not accounted for in FTS. Students have limitations on how many of these events they can do per day along with their flights so this model does not account for these constraints. Lastly, student warmup windows are not included. FTS does not ensure that students nearing a warmup window are scheduled a flight event to remain current. Furthermore, FTS is not rooted in the idea of *persistence*, so small changes to inputs could drastically change new schedules. A reformulation and the addition of new constraints can create a tool that is better equipped for creating schedules in accordance with winging student aviators.

Another example of an aviation-related scheduling problem is the work done by former NPS student Robert Slye for NAS Fallon. Slye created the Scheduling Assistance Tool (SAT) with the goal of scheduling training events under a variety of constraints to include airspace and aircraft availability (Slye 2018). The output of SAT is a “daily flight schedule that includes unit, event, day, start time, and range assignment,” but does not explicitly assign individual students and instructors (Slye 2018, p. v). It also does not follow any sort of syllabus flow or those accompanying requirements. However, it does operate under the ideas of *persistence*. The objective function incorporates penalties for canceled events, and the formulation includes a constraint that limits the number of cancellations.

Barkley, Foraker, Lazzaro, and Nelson (2019) use integer linear programming to create a tool that outputs schedules for U.S. Navy Strike Fighter Squadron 106 (VFA-106). Their goal is to maximize the number of events completed by students in a two-day period. Unlike VT-22’s problem, they incorporate air space along with numerous other constraints that are specific to the VFA-106 syllabus. Their integer linear program pairs students, instructors, events, planes, air spaces, and time. VFA-106 schedules events for three different categories of students: those who are re-qualifying, those who are qualifying for the first time, and those who are obtaining instructor qualifications. This model also takes into account aircraft composition by using variables that keep track of which person is in the backseat and which person is in the front seat for a given plane. The output is a two-day schedule with three periods per day. These periods or “waves” represent early morning, mid-morning, and afternoon (Barkley et al. 2019, p. 8). Scheduling events in just three periods does not give the resolution needed with respect to VT-22’s syllabus. While this model takes into consideration crew rest, it is missing several other aspects including non-availabilities, time specific flights, and student day limitations.

Lastly, Miller et al. (2017) use a mixed-integer linear program to optimally assign students of different types to one of the four Nuclear Propulsion Training Units for training in adherence with the U.S. Naval Nuclear Propulsion Training Program. Students are assigned in batches to one of the Nuclear Propulsion Training Units, and each batch of students is a class that will go through their training together. Each class represents students of a certain type such as electrician’s mate and machinist’s mate. The five different student types have different qualifications, so they are grouped by type. Miller et al. develop the Training Capability Model (TCM), a mixed-integer linear program that allocates a number of students of each type to each Nuclear Propulsion Training Unit. It also “prescribes weekly staff-instructor assignments” and “weekly student watchstanding and off-watch training” (Miller et al. 2017, p. 322). They incorporate *persistence* into both their objective function and constraints. Their persistent constraints establish “goals and limits on adherence to a desired partial solution” (Miller et al. 2017, p. 325). In their objective function, they use a previous allocation as a referenced target for new allocations. Thus, TCM yields “no or at most only small changes in the TCM student-allocation prescriptions” when a user makes small adjustments to the inputs (Miller et al. 2017, p. 325).

III. TRAINING EVENT SCHEDULING TOOL

In this chapter, we discuss the objective, goals, limitations, assumptions and formulation of our Training Event Scheduling Tool integer linear program.

A. OBJECTIVE

The objective of TEST is to assign students to instructors, periods, and required events. In our formulation, we implement *persistence* by having a primary objective to complete "desired" events. TEST's objective function is an aspiration constraint that imposes penalties for deviance from a desired schedule. Schedulers input all of the "desired" events they want each student to complete by the end of the day or week, which could be based on a prior TEST prescription and/or the VT-22 schedulers' scheduling experience. There is a penalty associated with "desired" events that are incomplete by the end of the horizon being considered (day or week). TEST also penalizes scheduling events in undesirable periods such as on the weekends. Instructor assignments are also slightly penalized to ensure assignment only when required. Additionally, a reward is given for scheduling additional "possible" events, which are events in the student's current stage that are not already complete. SKEDSOs determine which events fall into the "desired" and "possible" event lists for each student. A reward is also associated with scheduling "desired" events earlier in the week. In general, TEST attempts to schedule as many events as possible with the priority being all of the events in the "desired" list.

B. GOALS

Here, we outline our summary of the goals currently used by VT-22 when manually creating schedules based on our interactions with the schedulers. We describe these goals in three parts: student, instructor, and overall goals. We also explain how TEST models each of these goals.

1. Student Goals

Student goals consist of minimizing time to train, maintaining at least a given number of flights each week, and keeping current. The first goal, minimizing time to train, ensures that the U.S. Navy is meeting force readiness standards by training and winging aviators quickly. Pilots who earn their “Wings of Gold” upon completion of Phase II join the fleet and head to an FRS. The second goal of maintaining a given number of flights each week ensures that each SNA is progressing through the syllabus. Instructors are typically the limiting resource at VT-22, and because of this, it is important to distribute their time amongst all the students so that everyone has a chance to fly weekly. This leads to the last student goal of keeping current. Schedulers record the number of days since a student’s last flight. When a student enters the “yellow zone” of falling out of currency, the schedulers prioritize assigning a flight event to that student. Keeping students out of the warmup window is very important because students who exceed the maximum number of days since last flight need to complete a warmup event. Thus, instead of making steady progress through the syllabus, the students must backtrack and recomplete events for which they have fallen out of currency.

TEST models the first student goal of minimizing training time by granting a reward for scheduling events earlier in the week as well as a reward for scheduling additional events from the “possible” event list. Additionally, TEST imposes a penalty for not scheduling “desired” events that the schedulers determine are the events that each student should complete by the end of the week. All of these aspects help achieve minimal student training time. TEST models the second student goal of maintaining a given number of flights each week through the use of an elastic constraint (penalized if violated) that requires each SNA complete at least one flight event by the end of the week if they have any flights in their “possible” list. Lastly, TEST models the goal of keeping students current by allowing schedulers to impose a penalty for students not completing specific events. Thus, SKEDSOs have the ability to prioritize specific events for students who are nearing a warmup window.

2. Instructor Goals

Goals for instructors include scheduling an instructor to a minimum number of events per day and minimizing idle time. Because instructors are the bottleneck in scheduling, assigning them a minimum number of events per day allows for better utilization of limited resources. Currently, the Scheduling Office aims to assign two events to an instructor per day. Even this is a difficult task due to instructor qualifications and availabilities. In an effort to maintain good quality of life, schedulers constrain Friday schedules to 1800. Thus, they struggle to assign even two flights to instructors before 1800.

TEST assigns the correct number of instructors (to either teach or lead in the case of multiplane flight events) to every student in every event. While we can easily add another constraint to ensure a minimum number of instructor events per day, such a constraint could result in extra instructors unnecessarily being assigned to student events. Consequently, we penalize assigning instructors to events to ensure only the minimum number of instructors required for an event instructs. Instructors who do not instruct a SNA will utilize their time by undergoing instructor training to earn additional qualifications and retain currency of qualifications. Current TEST results indicate that TEST schedules instructors to two events per day where instructor availabilities permit. TEST has the added ability to "surge" events by assigning up to three flight events to instructors per day. Due to morale, this is not something that will occur daily. SKEDSOs and instructors decide which day of the week works well for a "surge" event.

The second instructor goal is minimizing idle time. This means minimizing the time between events in a given day for each instructor. In doing so, instructors would have longer contiguous rest periods because they would not have to wait for hours between flights. TEST does not currently address this goal. In our section on future research, we discuss proposals on how this can be achieved.

3. Overall Goals

Overall goals consist of minimizing the need for warmup events, creating robust schedules, and outputting schedules quickly in an easy-to-read format. Warmup events not only halt a student's progress through the syllabus but also require an instructor to fly these warmup events when they could have been utilized elsewhere. Because the squadron is limited by instructors, we want to minimize "storage cost," which means scheduling students to events in a timely manner, minimizing warmup events, and ultimately decreasing time to train. A robust schedule is one that minimizes disruptions when something does not go as planned. Exploring run time may be the solution to this. When something suddenly changes, schedulers need the ability to output a new schedule with new inputs in a matter of minutes so they can meet their goal of delivering the daily schedule at 1600.

TEST models minimizing the need for warmup events by giving schedulers the ability to prioritize specific events that a student needs to complete by a certain time. Schedulers can create one-day schedules with higher priorities for students nearing a warmup window to ensure those students complete all necessary events the following day. Some of the schedulers at VT-22 have over ten years of experience and a good idea of what it means to create a robust schedule. Some considerations for creating a robust schedule include identifying any events that students have failed in the past and scheduling these appropriately. Another consideration is not scheduling events at the beginning of the day that are prerequisites for events later in the day, so if a student fails the first event of the day, he or she can still fly the second event. Creating schedules manually is extremely time consuming. TEST creates a one-day schedule in a matter of minutes. Additionally, TEST outputs schedules in an easy-to-read format with the event name, time of day, and student and instructor names. Ultimately, the goal of this tool is to maximize sortie production and making winged aviators. The goals identified above and the explanations of how TEST models these goals are the key to achieving this.

C. LIMITATIONS

TEST has a few limitations that may serve as future work efforts:

- TEST is not integrated with TIMS, VT-22's current management system that publishes the daily schedules online, therefore schedulers must update TIMS manually with the output from TEST;
- TEST requires manual inputs that may be time-consuming to enter; and
- TEST does not consider lead instructors when assigning instructors in multiplane events. Instructors assigned in multiplane events have the basic qualifications required for that stage.

D. ASSUMPTIONS

The following is a list of assumptions we made when formulating TEST:

- We assume hourly time periods are appropriate because they provide sufficient resolution, are realistic, provide good solution times, and can be adjusted to 15-minute time periods (see our future research section);
- We assume aircraft maintenance does not affect scheduling flights;
- We assume crew rest always occurs from the end of one day and proceeds into the start of the next day; and
- We assume each student should be scheduled for at least one flight event during the week if there are any flight events in his or her "possible" list.

E. FORMULATION

1. Indices [~ cardinality]

D	The set of days: $d \in D = \{D1, D2, ...D7\}$ (ordered set) [7].
E	The set of all student events: $e \in E = \{CO3101, \ldots\}$ [258].
I	The set of instructors: $i \in I$ [32].
J	The set of stages: $j \in J = \{CO, EP, ...E2/C2\_CQ\}$ [24].
P	The set of periods: $p \in P = \{p1, p2, ...p168\}$ (ordered set) [168].
S	The set of students: $s \in S$ [122].

2. Subsets

$A_e \subset P$	Periods where event e is allowed to start.
$DD \subset E \times E$	$(e', e) \in DD$ if event e' cannot be the same day as event e.
$E^f \subset E$	Events that are flight events.
$E^j \subset E$	Events that are in stage j.
$E^{lec} \subset E$	Events that are lectures.

$E^{only} \subset E$	Events that can be scheduled only with lecture events occurring the same day.
$E^{onlyFlight} \subset E$	Flight events that can be scheduled only with lectures and simulator events occurring the same day.
$E^{NOW} \subset E$	Events that should not be flown with on-wing instructor.
$E^{OW} \subset E$	Events required to be flown with on-wing instructor.
$E^{sim} \subset E$	Events that are simulator events.
$I_e \subset I$	Instructors who can instruct event e.
$OW_s \subset I$	Instructors who are on-wing for student s.
$P_d \subset P$	Periods in day d (ordered subset).
$P_i^{NO} \subset P$	Periods where instructor i is not available.
$P_s^{NO} \subset P$	Periods where student s is not available.
$R \subset E \times E$	$(e', e) \in R$ if event e' precedes event e.
$S_e \subset E$	Events e' that satisfy event e.
$SP_{ep} \subset P$	All periods p' that prohibit starting any event if event e started in period p.

3. Data

$Buffer_e$	Time required after event e before starting the next event. [periods]
CAP	Maximum number of students that can be assigned to a lecture event in a given period. [students] Default $CAP = 20$.
$complete_{e,s}$	1 if student s already completed event e, 0 otherwise.
$CQwave_p$	Maximum number of students that can be waved during a CQ event in period p. [students]

$EEvent_{d,i}$	1 if instructor *i* can be scheduled for an extra event on day *d*. [events]
$G_{e,s}$	1 if requiring completion of event *e* for student *s* (desired), 0 otherwise.
instructorP	Penalty for scheduling instructors to events. [number] Default $instructorP = 0.01$.
M_p	Maximum number of flights in period *p* (jet availability). [aircraft] Default $M_p = 21\ \forall p \in P$.
NA_e	Number of aircraft required for event *e*. [aircraft]
NI_e	Number of instructors required for event *e*. [instructors]
NUM_e	Number of periods in event *e*. [periods]
$possible_{e,s}$	1 if student *s* can complete event *e* during the week, 0 otherwise. Calculated to be all events in $stage_{j,s}$ that are not in $complete_{e,s}$ where $stage_{j,s}$ is 1 if student s is in stage *j* and 0 otherwise.
$Reward_{e,p,s}$	Reward for scheduling student *s* to event *e* in period *p*. [number] Calculated as $scheduleR_{e,p,s} exR_p\ \forall e \in E, p \in P, s \in S$ where $scheduleR_{e,p,s}$ is the reward for scheduling student *s* to event *e* in period *p* [number] and exR_p is the reward for scheduling an event in period *p* that is earlier in the week, which is calculated as 0.99^{p-1}. [number]
$scheduleP_p$	Penalty for scheduling an event in period *p*. [number]
simCAP	Maximum number of students that can be in a simulator event in any period. [students] Default $simCAP = 5$.

$studentP_{e,s}$ Penalty for student *s* not completing event *e*. [number]

4. Decision Variables

$ELAS_s$ Non-negative variable with value of 1 if student *s* is not scheduled to a flight event in a week.

$L_{e,i,p}$ Binary variable with value of 1 if instructor *i* instructs lecture event *e* in period *p* and 0 otherwise.

$LEC_{d,s}$ Binary variable with value of 1 if student *s* has more than four lecture hours on day *d* and 0 otherwise.

$LECI_{d,i}$ Binary variable with value of 1 if instructor *i* instructs more than four lecture hours on day *d* and 0 otherwise.

$Y_{e,i,p}$ Binary variable with value of 1 if instructor *i* flies event *e* in period *p* and 0 otherwise.

$X_{e,p,s}$ Binary variable with value of 1 if student *s* starts event *e* in period *p* and 0 otherwise.

5. Objective

$$\begin{aligned} Min \sum_{e|G_{e,s}=1,s} (G_{e,s} studentP_{e,s} - \sum_{p \in A_e} X_{e,p,s}) + \sum_{e,p \in A_e,s} scheduleP_p X_{e,p,s} \\ + \sum_{e|possible_{e,s}=1, p \in A_e, s} Reward_{e,p,s} X_{e,p,s} + \sum_{e,i,p \in A_e} instructorP(Y_{e,i,p} + L_{e,i,p}) \\ + \sum_s flightP(ELAS_s) \end{aligned} \quad (0)$$

6. Constraints

$$X_{e,p,s} \le \sum_{p' \le p - NUM_{e'} - Buffer_{e'} \cap p' \in A_{e'}} X_{e',p',s} \qquad \begin{aligned} &\forall (e',e) \in R \,| \\ &possible_{e,s} = 1, \\ &complete_{e',s} = 0, \\ &possible_{e',s} = 1, p \in A_e, \\ &s \in S \end{aligned} \quad (1)$$

$$\sum_{p \in A_e} X_{e,p,s} \le 1 \qquad \forall e \in E \mid possible_{e,s} = 1, \; s \in S \qquad (2)$$

$$\sum_{e \mid possible_{e,s}=1,\, p-NUM_e-Buffer_e+1 \le p' \le p \cap p' \in A_e} X_{e,p',s} \le 1 \qquad \forall p \in P, s \in S \qquad (3)$$

$$\sum_{e,\, p-NUM_e-Buffer_e+1 \le p' \le p \cap p' \in A_e} (Y_{e,i,p'} + L_{e,i,p'}) \le 1 \qquad \forall i \in I, p \in P \qquad (4)$$

$$\sum_{e \in E^f \mid possible_{e,s}=1,\, p \in A_e} X_{e,p,s} \ge 1 - ELAS_s \qquad \forall s \in S \mid \sum_{e \in E^f} possible_{e,s} \ge 1 \qquad (5)$$

$$\sum_{e \in (E^{sim} \cup E^f) \cap e \notin (E^{CQ} \cup E^{only}) \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 2(1 - LEC_{d,s} - \sum_{e \in E^{only} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s}) \qquad \forall d \in D, s \in S \qquad (6)$$

$$\sum_{e \in E^f \cap e \notin (E^{CQ} \cup E^{onlyFlight}) \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 2(1 - \sum_{e \in E^{onlyFlight} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s}) \qquad \forall d \in D, s \in S \qquad (7)$$

$$\sum_{e \in E^{lec} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} NUM_e \le 4 + 4LEC_{d,s} \qquad \forall d \in D, s \in S \qquad (8)$$

$$\sum_{e \in E^{onlyFlight} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 1 - LEC_{d,s} \qquad \forall d \in D, s \in S \qquad (9)$$

$$\sum_{e \in E^{lec} \cap e \notin E^{only} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 8(1 - \sum_{e \in E^{only} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s}) \qquad \forall d \in D, s \in S \qquad (10)$$

$$\sum_{e \in E^{lec} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 8(2 - \sum_{e \in (E^f \cup E^{sim}) \cap e \notin E^{CQ} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s}) \qquad \forall d \in D, s \in S \qquad (11)$$

$$\sum_{e \in E^{lec} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s} \le 8(3 - \sum_{e \in E^{CQ} \cap e \notin E^{lec} \mid possible_{e,s}=1,\, p \in P_d \cap A_e} X_{e,p,s}) \qquad \forall d \in D, s \in S \qquad (12)$$

$$\sum_{e \in E^{CQ} \cap e \notin E^{lec} | possible_{e,s}=1, p \in P_d \cap A_e} X_{e,p,s} \leq 3(1 - LEC_{d,s}) \qquad \forall d \in D, s \in S \qquad (13)$$

$$\sum_{e' | possible_{e',s}=1, p' \in (SP_{e'p} \cap A_{e'})} X_{e',p',s} \leq 8(1 - \sum_{e | possible_{e,s}=1} X_{e,p,s}) \qquad \forall s \in S, p \in P \qquad (14)$$

$$\sum_{e \in E^f, p \in P_d \cap A_e} Y_{e,i,p} \leq (2 + EEvent_{d,i})(1 - LECI_{d,i}) \qquad \forall d \in D, i \in I \qquad (15)$$

$$\sum_{e \in E^{lec}, p \in P_d \cap A_e} L_{e,i,p} NUM_e \leq 4 + 4LECI_{d,i} \qquad \forall d \in D, i \in I \qquad (16)$$

$$\sum_{e', p' \in (SP_{e'p} \cap A_{e'})} (Y_{e',i,p'} + L_{e',i,p'}) \leq 8(1 - \sum_{e} (Y_{e,i,p} + L_{e,i,p}) \qquad \forall i \in I, p \in P \qquad (17)$$

$$\sum_{s | possible_{e,s}=1} X_{e,p,s} \leq \sum_{i \in I_e} L_{e,i,p} CAP \qquad \forall e \in E^{lec}, p \in A_e \qquad (18)$$

$$\sum_{e \in E^{CQ} \cap E^f, s | possible_{e,s}=1} X_{e,p,s} \leq CQwave_p \qquad p \in A_e \mid CQwave_p > 0 \qquad (19)$$

$$\sum_{e | possible_{e,s}=1, p - NUM_e + 1 \leq p' \leq p | p' \in A_e, s} NA_e X_{e,p',s} \leq M_p \qquad \forall p \in P \qquad (20)$$

$$\sum_{s | possible_{e,s}=1} NI_e X_{e,p,s} = \sum_{i \in I_e} Y_{e,i,p} \qquad \forall e \in E^f, p \in A_e \qquad (21)$$

$$\sum_{p \in P_d \cap A_e} X_{e,p,s} + \sum_{p \in P_d \cap A_{e'}} X_{e',p,s} \leq 1 \qquad \forall d \in D, (e', e) \in DD \mid (possible_{e,s} = 1, possible_{e',s} = 1, complete_{e',s} = 0, complete_{e,s} = 0), s \in S \qquad (22)$$

$$X_{e,p,s} \leq \sum_{i \in I_e \cap OW_s} Y_{e,i,p} \qquad \forall e \in E^{OW}, p \in A_e, s \in S \mid possible_{e,s} = 1 \qquad (23)$$

$$X_{e,p,s} \le \sum_{i \in I_e | i \notin OW_s} Y_{e,i,p} \qquad \forall e \in E^{NOW}, p \in A_e, s \in S \mid possible_{e,s} = 1 \qquad (24)$$

$$\sum_{e \in E^{OW} | p \in A_e, s | possible_{e,s} = 1 \cap i \in (I_e \cap OW_s)} X_{e,p,s} \le 1 \qquad \forall i \in I, p \in P \qquad (25)$$

$$\sum_{s, e \in E^{sim} | possible_{e,s} = 1, p - NUM_e + 1 \le p' \le p | p' \in A_e} X_{e,p',s} \le simCAP \qquad \forall p \in P \qquad (26)$$

$$\sum_{p - NUM_e + 1 \le p' \le p \cap p' \in A_e} (Y_{e,i,p'} + L_{e,i,p'}) = 0 \qquad \forall e \in E, i \in I, p \in P_i^{NO} \qquad (27)$$

$$\sum_{p - NUM_e + 1 \le p' \le p \cap p' \in A_e} X_{e,p',s} = 0 \qquad \forall e \in E, p \in P_s^{NO}, s \in S \qquad (28)$$

$$Y_{e,i,p}, L_{e,i,p} \in \{0,1\} \qquad \forall e \in E, i \in I, p \in P \qquad (29)$$

$$X_{e,p,s} \in \{0,1\} \qquad \forall e \in E, p \in P, s \in S \qquad (30)$$

$$LEC_{d,s} \in \{0,1\} \qquad \forall d \in D, s \in S \qquad (31)$$

$$LECI_{d,i} \in \{0,1\} \qquad \forall d \in D, i \in I \qquad (32)$$

$$ELAS_s \ge 0 \qquad \forall s \in S \qquad (33)$$

The objective function (0) expresses the cost of not scheduling events in the "desired" list, the penalties for scheduling in undesirable periods, assigning instructors to events, and not scheduling students to at least one flight event, and the reward for completing additional events.

Constraint (1) ensures adherence to precedence relationships between events in the syllabus.

Constraint (2) ensures each event is only completed at most once by each student.

Constraint (3) ensures each student is scheduled for at most one event in each period.

Constraint (4) ensures each instructor instructs at most one flight or lecture event in each period.

Constraint (5) ensures each student is scheduled for at least one flight event in the week if there is a flight event in his or her “possible” list or records a deviation from this requirement.

Constraint (6) requires that students complete no more than two flight or simulator events per day except for CQ events and events that should be only scheduled one per day.

Constraint (7) requires that students complete no more than two flight events per day except for when a student has a flight event that should not be completed with any other flight event.

Constraint (8) allows students to complete up to four lecture hours on days where they have a simulator or flight event and up to eight lecture hours if they do not have a simulator or flight event.

Constraint (9) ensures flight events that should only be completed one per day are not paired with more than four hours of lectures on the same day.

Constraint (10) ensures events that should only be completed one per day are not paired with any lecture events.

Constraint (11) ensures that if a student has two flight or simulator events per day, he or she should not be scheduled for any lecture events that day.

Constraint (12) ensures that if a student has three flight or simulator events per day while in the CQ stage, he or she should not be scheduled for any lecture events that day.

Constraint (13) allows students to complete up to three CQ flight or simulator events per day if they do not have more than four lecture hours on the same day.

Constraint (14) ensures adherence to student crew day limitations.

Constraint (15) requires that instructors teach no more than two flight or simulator events per day unless they have a “surge” event.

Constraint (16) allows instructors to teach up to four lecture hours on days where they fly and up to eight lecture hours if they do not have a flight event.

Constraint (17) ensures adherence to instructor crew day limitations.

Constraint (18) ensures instructors are scheduled to instruct lecture events while adhering to classroom capacities.

Constraint (19) allows LSOs to wave up to a certain number of students in a given period.

Constraint (20) ensures adherence to jet availability.

Constraint (21) ensures the correct number of instructors are assigned to fly each event.

Constraint (22) ensures certain events are scheduled on different days.

Constraint (23) requires students fly with their on-wing for on-wing events.

Constraint (24) requires students do not fly with their on-wing for non on-wing events.

Constraint (25) ensures that if multiple students share an on-wing instructor, they do not start the on-wing event in the same period.

Constraint (26) restricts the number of students assigned to simulator events in a given period.

Constraint (27) and (28) adheres to instructor and student non-availabilities.

Constraints (29) to (33) declare variable types.

F. SUMMARY

TEST outputs a daily or weekly schedule with student, instructor, event, and period assignments. TEST minimizes deviance from what the schedulers would like the student to complete. The schedule outputs to a CSV file where students and instructors can see each day's scheduled events. The output also displays the length of each event as well as the corresponding time for each period.

IV. TESTING TEST

This chapter covers the implementation of the Training Event Scheduling Tool, the real-world data collected from VT-22 to use as input to TEST, our model's output, and a comparison between TEST and the manually created schedules.

A. IMPLEMENTATION

TEST uses Python version 3.6 and the Python-based optimization modeling language Pyomo, which is open-source for easy distribution (Hart et al. 2012). CPLEX 12.8 solves TEST (IBM 2017). We implemented TEST on a 64-bit Dell Latitude E6440 with two 2.90GHz processors and 8GB of RAM. Using data provided by VT-22, a daily instance takes less than ten minutes to solve to a gap of less than 1%. There are about 218,000 constraints and 516,000 binary variables for the daily schedule. A weekly instance can solve in about three hours and consists of 565,000 constraints and 3,300,000 binary variables. Gaps for the weekly schedule vary greatly depending on the inputs. In one instance, TEST requires five hours to achieve a 10% gap. In another, TEST ran for seven hours and achieved a 68% gap. Even with this large reported gap, we find TEST results are significantly better in scheduling more events compared to their manual counterpart.

B. DATA COLLECTION

Over the course of a week, VT-22 collected data to be used as inputs to TEST. From September 30, 2019 to October 4, 2019, VT-22 schedulers recorded and provided instructor qualifications, student non-availabilities, instructor non-availabilities, students' current completion, and jet availability. Due to some students going on detachment during this week, the total number of students at VT-22 was 66. Additionally, there were 45 instructors who could instruct events during this week. Some of these instructors are not usually attached to VT-22 but were able to accommodate flights in specific time periods over the course of the week. We used the manually created schedules posted online to determine which events we desire each student to complete by the end of the week (Chief of Naval Air Training 2019). VT-22 also provided the schedule results which includes the flight events, simulator events, and lecture events that were or were not completed each day.

C. MODEL INPUTS

TEST reads in one large Excel spreadsheet with multiple sheets that represent all of the model inputs. Most sheets remain consistent from week to week. Schedulers need to update other sheets manually each time they run TEST. The list of students and instructors may change depending on whether a new group of students reports to the squadron, students graduate from the program, or additional instructors who are not normally attached to VT-22 have availability to fly VT-22 events. SKEDSOs fill out the "desired" events for each student, which are all the events they want the student to complete by the end of the week (Table 1). A "1" indicates the student should complete the event, and a "0" or no input indicates it is not a "desired" event.

Table 1. Sample desired event input.

Student	CO3101	CO3102	CO3201	CO3202	CO1106-07	EP2101	EP2102	EP2103
ENS X	1	1	1					
LTJG Y				1	1	1	1	
1stLT Z						1	1	1

If schedulers would like a SNA to complete events in a stage that should not be completed alongside any other stage, they should not include events from other stages in the "desired" event list. Schedulers should only input "desired" events that a student can complete based on his or her completed events and the prerequisites. Schedulers also need to input all the events that each student already completed (Table 2).

Table 2. Sample completed event input.

Student	CO3101	CO3102	CO3201	CO3202	CO1106-07	EP2101	EP2102	EP2103
ENS X								
LTJG Y	1	1	1					
1stLT Z	1	1	1	1	1			

SKEDSOs need to update all the stages for each student (Table 3). The "possible" event list includes all of the "desired" events as well as all the remaining events in the stages a student is in if these events are not already complete. For example, if the schedulers want a student to proceed to events in the Strike stage after all the events in the Operational Navigation stage are complete, they input the desired Operational Navigation events in the "desired" list and include Strike in the stage list for that student.

Table 3. Sample stages for each student input.

Student	CO	EP	BI	RI	FAM/OCF
ENS X	1				
LTJG Y	1	1			
1st LT Z		1	1		

Schedulers also have the ability to schedule an additional flight event to each instructor over the course of the week (Table 4). Currently, instructors can have at most two flight events per day, but schedulers have the ability to "surge" events on days that work well for individual instructors.

Table 4. Sample instructor availability to fly additional flights for each day input.

Instructor	D1	D2	D3	D4	D5	D6	D7
Maj A	1						
LT B				1			
LtCol C		1					

Schedulers should regularly update instructor and student non-availabilities (Table 5). A "1" indicates that an instructor cannot be assigned any event in that particular period. TEST accounts for length of events to ensure that scheduled events in prior periods do not violate these non-availabilities for both students and instructors.

Table 5. Sample instructor non-availabilities input.

Instructor	P7 (0700)	P8 (0800)	P9 (0900)	P10 (1000)	P11 (1100)	P12 (1200)	P13 (1300)	P14 (1400)
Maj A	1	1						
LT B								
LtCol C					1	1	1	

The next inputs do not require regular updating unless syllabus requirements change or instructors obtain new qualifications. TEST takes into account the number of periods (Table 6), number of instructors, and number of aircraft required for each event.

Table 6. Sample number of periods for each event input.

CO3101	CO3102	CO3201	CO3202	CO1106-07	EP2101	EP2102	EP2103
3	3	3	3	2	3	3	3

The FIST, which tracks instructor qualifications in each stage, is another input to TEST (Table 7). Simulator events and some lecture events do not have any instructor pilots listed because these events are taught by civilian personnel.

Table 7. Sample instructor qualifications input.

IR4101	IR4102	IR4290	FRM1101-06	FRM3101	FRM3102	FRM4101	FRM4102
Maj A	Maj A	Maj A				LtCol C	LtCol C
LT B	LT B	LT B					
LtCol C	LtCol C	LtCol C					

The prerequisite matrix depicts the immediate prerequisite events that a student must complete prior to an event assignment (Table 8). The prerequisite matrix shows how the student flow in Phase I is mostly linear and how students have the ability to "jump around" in Phase II. In general, students complete lectures at the beginning of each stage before conducting any simulator or flight events. Additionally, students cannot start Phase II without finishing all events in Phase I.

TEST accommodates both Strike and E2/C2 syllabi. SNA assigned to the E2/C2 syllabus complete most of the events in Phase 1 and then complete some additional events that are specific to the E2/C2 track. Schedulers use TEST to distinguish E2/C2 and Strike students by placing E2/C2 students in the E2/C2 stage. TEST also ensures that E2/C2 specific events are not prerequisites for other Strike events. TEST assigns E2/C2 students to specific E2/C2 events once all the prerequisites have been met.

Table 8. Sample prerequisite matrix input.

	CO3101	**CO3102**	**CO3201**	**CO3202**	**CO1106-07**	**EP2101**	**EP2102**	**EP2103**
CO3101		1						
CO3102			1					
CO3201				1				
CO3202					1	1		
CO1106-07								
EP2101							1	
EP2102								1
EP2103								

TEST takes into account all the periods in which every event can occur (Table 9). This matrix does not incorporate event lengths as that is handled separately in TEST. Below we list start and end times for results reported in this thesis. Simulator events occur between 0600 and 2200. Flight events occur depending on whether they are a day flight, a night flight, or either a day or night flight. Briefing for day flights begins at 0800 and debrief finishes by 1900. Night flights occur between 1800 and 2400. Flight events that can be completed as either day or night encompass both day and night times. Lecture events occur between 0700 and 1700.

Table 9. Sample periods during which an event can occur input.

	CO3101	CO3102	CO3201	CO3202	CO1106-07	EP2101	EP2102	EP2103
P1								
P2								
P3								
P4								
P5								
P6	1	1	1	1		1	1	1
P7	1	1	1	1	1	1	1	1
P8	1	1	1	1	1	1	1	1

If students are in the CQ stage and need to conduct CQ flight events, schedulers should adjust $CQwave_p$, the number of students that an LSO can wave in a given period. LSOs can wave up to five students at night and up to six during the day. SKEDSOs also need to make LSOs unavailable in the Instructor Non-Availability matrix for all the time periods they would like the LSOs to wave students. This ensures they do not fly any other event and are available to stand by the runway and wave groups of students in the CQ stage.

Lastly, schedulers can update the objective function penalties and rewards. Below we list the penalties for results reported here. The student penalty for not completing one of the “desired” events is set to 10. The schedule penalty is the penalty for scheduling in certain periods. It is 0 during the week and 20 for scheduling on the weekends. The objective function also imposes a small penalty for scheduling instructors to events to ensure they are not scheduled unnecessarily. This value is set to 0.01. TEST gives a reward for scheduling additional events in the “possible” list. This default value is -5. The penalty for not scheduling “desired” events should be greater than the reward for scheduling additional “possible” events. TEST also rewards scheduling events earlier in the time horizon being considered (daily or weekly) using an exponential of 0.99 raised to the hourly period minus one. Lastly, TEST imposes a penalty for not scheduling a student to a flight event if he or she had a flight event in the “possible” list. This value is set to 15.

D. OUTPUT

TEST outputs the schedule to a CSV file in Excel. This file includes either a daily or weekly schedule (Table 10). For each day, the student names appear first in the leftmost column followed below by the instructor names in alphabetical order. To the right of the student and instructor names are the event names and type. The schedule displays the length of each event and corresponding period.

Table 10. Sample output from TEST for one day.

Day 1	P8:0800	P9:0900	P10:1000	P11:1100	P12:1200	P13:1300	P14:1400
ENS X	F:BI4101	F:BI4101	F:BI4101	F:BI4101	F:BI4101		
LTJG Y	S:OCF3101	S:OCF3101	S:OCF3101	F:FAM4304	F:FAM4304	F:FAM4304	F:FAM4304
1stLT Z	L:FAM1104			F:FAM4301	F:FAM4301	F:FAM4301	F:FAM4301
Maj A	F:BI4101	F:BI4101	F:BI4101	F:BI4101	F:BI4101		
LT B				F:FAM4301	F:FAM4301	F:FAM4301	F:FAM4301
LtCol C				F:FAM4304	F:FAM4304	F:FAM4304	F:FAM4304

The complete schedule includes all 24 periods representing hours in each day. Table 10 displays only periods in the middle of the day from 0800 to 1400. Flight event names are preceded by “F,” simulator event names are preceded by “S,” and lecture event names are preceded by “L.” TEST ensures instructor assignments for all flight events that require an instructor. In this example, the lecture and simulator events are not taught by a military instructor, so TEST does not assign one.

E. COMPARISON

The main metric we use to compare TEST schedules and those manually created by VT-22 is the total number of scheduled events. First, we ran TEST for seven days assuming good weather conditions and compared the number of scheduled events to those scheduled manually for each day (Table 11). This should be considered an optimistic view of what is possible because it does not account for unavailable periods due for example to weather.

Table 11. Number of scheduled events by day for current VT-22 schedules compared to TEST assuming good weather.

Day	VT-22 schedules	TEST with good weather
1	51	69
2	59	81
3	58	76
4	59	70
5	54	75
Total	281	371

From this, we see that on all days, TEST assigns more events. Days 1 and 2 were not affected by weather, and even in these days, TEST assigns a total of 40 more events. The schedules adhere to crew day, instructor qualifications, prerequisites, and non-availabilities. Next, we ran TEST for seven days with known bad weather and compared the results to the manually created schedules (Table 12). Using the schedule results provided by VT-22, we made all instructors unavailable to fly events in periods that were affected by bad weather or bird activity.

Table 12. Number of scheduled events by day for current VT-22 schedules compared to TEST assuming bad weather.

Day	VT-22 Schedules	TEST with bad weather
1	51	69
2	59	81
3	58	72
4	59	45
5	54	73
Total	281	340

Even when subject to bad weather, TEST schedules more events. During bad weather periods, TEST schedules students to more simulator and lecture events since these can be completed independent of the weather. The breakdown of type of events by day provides some further insight.

TEST schedules the majority of lecture events in the beginning of the week (Table 13). This is most likely due to the fact that lecture events are typically shorter than flight and simulator events. TEST rewards scheduling events earlier in the week and thus, TEST can fit in more events earlier in the week if it schedules lecture events. By the end of the week, TEST schedules only a few lecture events because there are a limited number of lecture events remaining in each student's "possible" list. Overall, TEST schedules more lecture events than the manually created schedules even with bad weather.

Table 13. Number of lecture events scheduled by day.

Day	VT-22: Lecture Events	TEST with bad weather: Lecture Events
1	5	19
2	8	15
3	10	12
4	14	3
5	0	5
Total	37	54

TEST is consistent with scheduling the same number of simulator events each day even with bad weather (Table 14). TEST schedules a much larger number of simulator events compared to the manually created schedules. However, SKEDSOs do not have the simulator instructors' availability days in advance. Schedules from TEST will need to be coordinated daily with the simulator instructors to ensure there are available instructors.

Table 14. Number of simulator events scheduled by day.

Day	VT-22: Simulator Events	TEST with bad weather: Simulator Events
1	16	27
2	20	26
3	16	26
4	14	25
5	15	25
Total	81	129

Even when subject to the same bad weather periods as the manually created schedules, TEST schedules almost the same number of flight events as VT-22 (Table 15). With perfect weather in the first two days, the number of flight events TEST schedules is almost exactly the same as the number of events VT-22 scheduled manually. TEST is on par with the manually created schedules for scheduling flight events when there is perfect weather but is able to schedule significantly more lecture and simulator events. Compared to how many flights were actually completed, TEST schedules slightly more flight events in the days that were unaffected by weather and significantly more flights in total. It could be argued that TEST has an unfair advantage due to the fact that it has perfect weather information and knows exactly which periods need to be blocked out. While this may slightly skew results, it shows the value of using weather forecasts to predict which periods may become unavailable for flying so that simulator and lecture events can be more efficiently planned.

Table 15. Number of flight events scheduled and completed by day.

Day	VT-22: Scheduled Flight Events	VT-22: Completed Flight Events	TEST with bad weather: Flight events
1	30	29	23
2	31	30	40
3	32	25	34
4	31	8	17
5	39	25	43
Total	163	117	157

There are a few other considerations when comparing the schedules created by VT-22 and the schedules outputted by TEST. TEST is conservative when assigning instructors to students because TEST does not combine student events that can share instructors to meet multiplane requirements (see future research section). Additionally, TEST does not utilize out-and-in events to decrease event times by assigning the same instructor-student pair in two different events (see future research section). Utilizing out-and-in events would remove the hour long buffer that is required between events and would decrease pre-brief and debrief times. Lastly, TEST assigns at least one event to every SNA over the course of the week. In the actual schedules, some students were not scheduled for any events during the week. Ensuring students complete at least one event potentially decreases the number of students who fall out of currency and have to perform warmup events.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION

TEST can successfully assist VT-22 schedulers in creating more efficient schedules that accommodate their goals. Compared to the manually created schedules, TEST assigns more lecture events, simulator events, and flight events to students. TEST prescribes scheduling assignments to all military instructors and students in both the E2/C2 and Strike syllabus. TEST adheres to a variety of constraints to include prerequisites for events in the Intermediate and Advanced phases, crew day and crew rest, non-availabilities, daytime and nighttime events, specific student-instructor pairs, jet availability, and simulator capacities. TEST requires schedulers to manually adjust inputs in the Excel sheets. The majority of these inputs will remain consistent from week to week such as prerequisites, penalties, number of periods in an event, and number of instructors in an event. Other inputs such as non-availabilities, “desired” events, and completed events require more frequent updates. SKEDSOs can run TEST to produce either a one-day schedule or a weeklong schedule. While a weeklong schedule can run for several hours, an optimal one-day schedule can output in a matter of minutes. Because TEST can rapidly re-construct high quality schedules, schedulers can use TEST to improve the next day’s schedule before they publish it based on new information received later in the day.

A. FUTURE RESEARCH

This thesis shows the value of TEST, but there are potential enhancements. Here we explore possibilities for future research such as implementing out-and-in events, creating a second pass model, and refining instructor assignments. The majority of these efforts can easily build on TEST while others require more time and additional resources.

1. Out-and-In Events

Out-and-in events are pairs of events in which the same instructor and student combination flies together. Instructor-student pairs can complete these events consecutively as long as the crew remains the same. Completing two events as an out-and-in event truncates the brief of the second flight by up to two hours so the entire pre-brief and debrief for both flights is approximately three hours. Typically, the first flight occurs during the daytime, and the second flight starts after sunset so that students meet their night hour minimums. Not utilizing out-and-in events by scheduling a new instructor in the second event of a pair not only increases briefing times but also requires students be given a one-hour break between flights.

TEST does not currently utilize out-and-in events. One can easily add out-and-in events to the event list and add a substitutability for these events that permits completion of the event to be satisfied individually or by the paired out-and-in event.

2. Training Integration and Management System and User Interface

TEST does not currently integrate with TIMS. TIMS provides non-availabilities for all students and instructors, which includes crew rest and squadron meetings. There are over 100 students checked into the squadron at one time. Integration with TIMS would greatly decrease the time required to input non-availabilities for each student as well as all of the instructors. Outputting schedules to TIMS would also eliminate the time it takes to input the manually created schedules that post online for the squadron to see.

Similarly, a user interface for TEST would drastically decrease the time it takes to manually input data. A user interface would also decrease the probability of human error when it comes to inputs for the many students and instructors at VT-22.

3. Second Pass Model

A second pass model would be very beneficial for refining schedules after TEST is run. This model would take as input the output from TEST and maintain most of the primary assignments. A pass through a second model could accommodate other scheduling needs that are not addressed in TEST such as a higher resolution of 15 minutes. A schedule with 15 minute periods would more closely resemble VT-22’s current schedules as flights take off in 15 minute intervals. This second pass model might also address the goal of minimizing instructor idle time as TEST does not specifically address this. A second pass model can impose a penalty for longer time periods between two instructor events. Minimizing idle times would allow instructors to have longer contiguous off periods. A second pass model might also ensure a robust schedule by taking into account events that students historically struggle with and not assigning a second event that depends on the completion of the first event in the same day. Lastly, a second pass model might take into account student start times so that one student is not bouncing between day and night flights over the course of the week.

4. Multiplane Event Combinations

Some events can be flown with two students and the same instructors to satisfy multiplane requirements. For example, SEM4103 requires three instructors and three planes. In this case, TEST would assign three different instructors to just one student to satisfy the three plane requirement. However, a second student can complete SEM4103 and utilize the same three instructors. TEST is conservative with instructor assignments, but since instructors are usually the limiting resource, TEST would be able to assign even more events if multiplane event combinations are considered.

5. Instructor Lead Qualifications

Refinement of instructor qualifications in TEST would enhance instructor assignments. For example, some instructors are qualified to instruct certain events while some are qualified to lead events. Currently, TEST does not make this distinction. One can easily implement this change by creating a separate list for each stage of all instructors who are qualified to lead events in that stage and ensuring that if an event requires a lead, TEST assigns one of these lead instructors.

6. Additional Resources and Syllabus Changes

Lastly, future research includes exploring a variety of questions that relate to total student time to train. These questions include:

- What if the squadron had more instructors or more students?,
- What if we assign more flights per instructors?, and
- What if we make syllabus changes by removing events?

An aggregate model that looks at a longer period (say weekly) of time over a year may help answer how these additions and adjustments alter total student time to train. Significant results may lead to permanent syllabus changes and squadron resource adjustments.

B. RECOMMENDATIONS

We recommend schedulers run TEST over the weekend for a long period of time to get a good solution for the weekly instance. Weather, bird activity, and aircraft maintenance are just some of the reasons canceled or incomplete flights occur. Oftentimes schedulers cannot predict these instances, and thus, the weekly schedule becomes less useful because events scheduled in the future usually depend on the completion of those scheduled at the beginning of the week. For this reason, we recommend SKEDSOs run TEST daily to accommodate last minute changes. TEST's quick run time in the daily instance makes this possible.

With TEST, schedulers have the ability to quickly generate efficient schedules that would otherwise take multiple hours to produce. TEST prescribes assignments of students, instructors, events, and periods that help achieve many of VT-22's current goals. Ultimately, TEST can aid VT-22 schedulers in creating schedules that decrease student time to train so that the U.S. Navy can enhance pilot training.

LIST OF REFERENCES

Barkley P, Foraker J, Lazzaro G, Nelson P (2019) Optimal Scheduling of Student and Instructor Aircrew for United States Navy Strike Fighter Squadron 106.

Brown GG, Dell RF, Wood RK (1997) Optimization and persistence. *Interfaces* 27(5):16– 18.

Chief of Naval Air Training (2019) Web Schedules. Accessed October 14, 2019, https://www.cnatra.navy.mil/scheds/schedule_data.aspx?sq=vt-22.

Goodwin K (2018) VT-22 scheduling via personal communication, November 2018.

Hart, W. E., Laird, C., Watson, J.-P., & Woodruff, D. L. (2012). *Pyomo—Optimization Modeling in Python*. New York, NY: Springer New York.

IBM (2017) IBM ILOG CPLEX 12.8 User's Manual (IBM ILOG CPLEX Division, Incline Village, NV).

Jacobs R (2014) *Optimization of Daily Flight Training Schedules*. Master's thesis, Naval Postgraduate School.

Miller MR, Alexander RJ, Arbige VA, Dell RF, Kremer SR, McClune BP, Oppenlander JE, Tomlin JP (2017) Optimal Allocation of Students to Naval Nuclear-Power Training Units. *Interfaces* 47(4): 320–335.

Naval Air Training Command (2014a) *T-45 E-2/C-2 Advanced Flight Training Curriculum*. Naval Air Station Corpus Christi, Corpus Christi, TX. https://www.cnatra.navy.mil/local/docs/instructions/1542.176.pdf.

Naval Air Training Command (2014b) *T-45 Combined Multi-Service Pilot Training System*. Naval Air Station Corpus Christi, Corpus Christi, TX. https://www.cnatra.navy.mil/local/docs/instructions/1542.167.pdf.

Naval Air Training Command (2017) Training Improvement Program (TIP). Naval Air Station Corpus Christi, Corpus Christi, TX. https://www.cnatra.navy.mil/local/docs/instructions/1550.6.pdf.

Simpson S (2019) VT-22 scheduling via personal communication, January 2019.

Slye R (2018) *Optimizing Training Event Schedules at Naval Air Station Fallon*. Master's thesis, Naval Postgraduate School.

U.S. Navy (2018) Commander Navy Installations Command Naval Air Station Kingsville. https://www.cnic.navy.mil/regions/cnrse/installations/nas_kingsville/about/tenant_commands/training_air_wing_two/flight_training.html.

VT-22 Golden Eagles (2018) In *Facebook* [Group page]. Retrieved November 7, 2019, from https://www.facebook.com/189731681196754/photos/p.1094061644097082/1094061644097082/?type=1&theater.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California