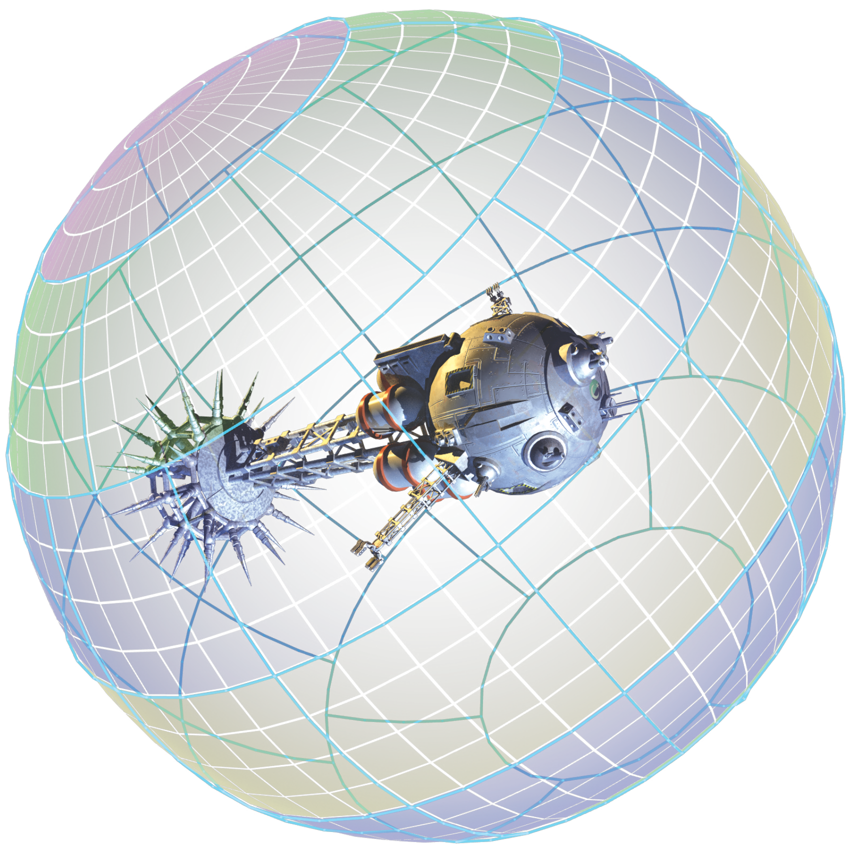
Attack Vector TACTICAL

Second Edition

By Ken Burnside, Eric Finley & Tony Valle



Copyright & Publishers Information

Published by:
Ad Astra Games
P O Box 389
Pelican Rapids, MN 56572
(218) 863 1784
adastragames@gmail.com

Rules questions are answered on the forums at http://www.adastragames.com/

Be sure to register your game at http://services.adastragames.com

Printing of 20 Feb 2011

Credits:

Cover Art: Joshua Qualtieri, Charles Oines, Ken Burnside & Winchell Chung. Composition by Philip Markgraf and Ken Burnside

Interior Art & Box Minis: Stephen Rider & Charles Oines

Game Design: Ken Burnside, Eric Finley & Tony Valle

Proofreading: Ethan McKinney, Leonard Conklin, Scott Palter

Layout, 2nd Edition: Ken Burnside

Layout, 1st Edition: Matt Arnold

More detailed descriptions of who contributed what to this project can be found in rule (Y1.0).

A

Don't Panic!

Attack Vector is a game with a lot of complex math underlying it. There's no two ways about it; we know, we did all of it to make the game work. This means you don't have to.

A lot of AV:T's rules are going to be different from any other space combat game you've seen. These differences exist because they're the only way to make the game work as both a representation of what we want it to do, and be playable by human beings without a computer assisting. In particular, there are places where we describe a procedure, and don't explain what the procedure does. The reason for this is because trying to explain the procedure and what the procedure is simulating would result in glazed over eyes.

In a lot of ways, AV:T is like a Swiss watch. The pieces and parts mesh together seamlessly, even if you don't know what, exactly, they're all doing.

A Note On Complexity

Games have three kinds of complexity:

1) Conceptual Complexity: How many new concepts do you have to learn to play? AV:T is High in this kind of complexity.

2) Procedural Complexity: How much record keeping is needed to play the game, and how common are rules exceptions? AV:T is Medium in this kind of complexity.

3) Decision Complexity: How many decisions are made during the game, and how many options must be kept in mind? AV:T is High in this kind of complexity.

In some ways, this is the 'good' kind of complexity, as it adds replay value.

(A1.0) INTRODUCTION

Attack Vector: Tactical (AV:T hereafter) is a wargame about spaceship combat played on a map of hexagons, or "hex grid." The players take the roles of warship commanders in combat, employing tactics, attempting to meet mission objectives, and operating within the constraints of their ships' capabilities.

AV:T's spaceships maneuver in accordance with Newtonian mechanics, using thrust and momentum to change their course. Once a ship is moving in a given direction, it will continue to move in that direction until it applies thrust in the opposite direction.

AV:T is a paper and pencil wargame in an era where computers can do impressive modeling and polygon counts. While a computer can do the math needed to run a space combat game, it does not help the player understand everything that is going on.

Moreover, computer games can rarely match the thrill of beating a live opponent by making better decisions and out-thinking them. Our goal in designing AV:T was to match tactical play with a good physics model, without sacrificing playability in the process.

To accomplish this, AV:T tucks a great deal of detail "under the carpet" of the rules. While we have taken every precaution to ensure accuracy, the rules present the results of time spent with equations, rather than the equations themselves. Where space permitted, we have included the background science behind the rule.

A typical game of AV:T takes roughly 20-30 minutes per turn; less in the hands of experienced players. A standard game takes somewhere between 7 and 15 turns to complete.

(A2.0) GENERAL COURSE OF PLAY

Games are played in scenarios, which provide the units and the objectives. Some scenarios may have secret objectives determined by drawing a card from a standard deck of playing cards. A scenario may be pre-provided, balanced by equal points, generated by a campaign game, or agreed on by the players. Each scenario is played in turns. Each turn is broken into 8 segments.

Play continues until one side has no chance of winning and attempts to disengage, surrenders, or is completely destroyed.

(A2.1) BASIC CONCEPTS:

Four concepts are fundamental to understanding AV:T: thrust, segmented movement, power management, and the Ship's Systems Display (SSD).

- Thrust creates acceleration and displacement, while expending fuel. Acceleration is the change in velocity in a given direction. Displacement is the actual movement of the ship while it accelerates.
- Segmented movement breaks a unit's movement down from hexes per turn to hexes per segment, allocating the movements proportionally. For example, a ship moving 7 hexes per turn would cover 7 hexes in 8 segments. Segmented movement tells you which segments the ship moves on. (In this particular example, the ship would move one hex on segments 2, 3, 4, 5, 6, 7 and 8). A ship moving 12 hexes per turn would move 1 hex on segment 1, 2 hexes on segment 2, 1 hex on segment 3, 2 hexes on segment 4, 1 hex on segment 5, 2 hexes on segment 6, 1 hex on segment 7 and 2 hexes on segment 8. A ship's record of how many hexes to move on which segment (and in which direction) is called the movement grid.
- Power management is the process of spending power from batteries, refilling the batteries from reactors, and storing the waste heat from reactors in heat sinks. The dynamic of power accumulation versus weapon cycle times sets the tempo of combat, while the ability to store heat in heat sinks sets the clock on the engagement.
- SSDs describe the capabilities of ships and have check boxes and spaces for recording damage to various systems. As ships take damage, systems are marked off as destroyed and the ships' capabilities decline.

(A2.2) USING WEAPONS:

There are two broad categories of weapons in the game: beam weapons and seeking weapons. To make an analogy, a policeman has a pistol and a dog. The pistol is a beam weapon, since the target cannot outrun the bullet. The dog is a seeking weapon; it's slower than the bullet, but can change course to intercept the target.

AV:T resolves weapons fire with rolls of a 10-sided die (or "d10"). Beam weapons have tables where you cross reference the die roll with the range bracket to find out how much damage was done. See the sidebar on this page.

3 MRLS — RANGE > / DIE ROLL	0-10	11-13	14-15	16-17	18-22	23-30
3(+5) 1	5	D7	D5	D5	D3	D1
2	6	2	D7	D7	D5	D3
3	6	3	1	1	D7	D5
4	7	4	2	1	1	D7
5	7	4	3	2	1	1
6	7	5	3	2	2	1
7	7	5	4	3	2	1
8	7	6	4	3	2	1
9	7	6	4	3	2	1
10	7	6	4	3	2	1

Reading A Weapon Table

A target is at range 10, and the die roll is 6. The first column shows a range of 0 to 10, so the target is in the first range bracket. A die roll of 6 means we read on the 6th row down, which nets 7 points of damage to the target.

A

(A2.3) TRUISMS OF SPACE COMBAT:

All players bring their own set of assumptions to a space combat game, either from television programs, popular fiction, or from other wargames of similar theme. Many of these assumptions don't apply to AV:T, and a number of the more common questions are addressed below to answer the questions of "where are they?"

STEALTH DOESN'T WORK: Space is vast, but mostly empty. Space is also dark and cold; the average background temperature of space is 2-5 Kelvin. Ships with habitable life support sections, even with the engines off, will have a surface temperature of at least 200 to 250 Kelvin (ice melts at 273 Kelvin). For a typical habitable section of a ship, the radiated heat signature is in the range of a few hundred kilowatts, which is generally detectable out to 30,000 km in under a day using a full spherical search pattern with a broad-field IR-band telescope with an aperture of 3 meters. While there is ecliptic dust that can re-radiate solar energy at about that temperature, the signal is much more dispersed, and likely doesn't show proper motion against the sky.

In addition to the waste heat generated by life support, a ship's power generation system generates heat. A perfect Carnot heat engine produces 2 watts of waste heat for every watt of electricity it produces, where waste heat dissipation is free (like in an atmosphere). In space, waste heat has to be radiated. Minimizing radiator size (to make them retractable in combat, and to make them mass less) means running them at a higher temperature, which reduces the efficiency of the Carnot cycle. Each radiator in AV:T is roughly a 25m x 25m surface radiating from both sides at around 1600 Kelvin. Each radiator disposes of roughly 44 GJ of waste heat in 128 seconds, for a signature strength of roughly 340 megawatts, which is detectable (easily) out to around 10 light seconds (3 million kilometers) under the same conditions as the crew's waste heat. (The distance from the Earth to the Sun is 500 light seconds, as a point of comparison.)

Beyond that, for a 5,000 ton ship using a reaction drive, even in cruise mode, it's producing a minimum of a 340 gigawatt signature at about 2800 K, which gives a 1 day spherical search pattern "guaranteed" detection radius of a bit over 1,000 light seconds, or roughly 2 AU. In low thrust fuel economy mode, it takes roughly 10-16 weeks to cross 1 AU. During this entire time, the people attempting detection need only look for a 14th magnitude star with measurable proper motion. (A ship in combat thrust puts out drive signatures 10 times as bright, and would be easily detectable out at roughly triple the ranges listed above, or around 6 AU).

DECOYS DON'T WORK: Any ship using a reaction drive reveals its mass by the correlation between observed rate of thrust and the temperature and brightness/mass spectroscopy of the exhaust plume. This means that unless the decoy weighs as much as the ship it's protecting and has the same drive as the ship its detecting, it will be be easy to distinguish against the sky.

Arguing Thermodynamics

The discussion about stealth in space is one that comes up periodically on the Ad Astra forums, and on the SFCONSIM-L mailing list; enough so that it's regularly met with eye rolling. Common attempts to justify stealth in space include:

Directional Radiation: This technique requires directing your thermal signature away from enemy sensors and coasting in 'running silent'. Aside from the multi-week (often multiple month) travel times it requires, it also requires knowing where the observation platforms are in the system, and is still voided whenever you apply thrust.

Hiding Behind a Soletta: A soletta is the spacegoing equivalent of a parasol, a big thin film of something meant to shade an object from solar radiation. When used for stealth, the soletta will pick up and re-radiate your thermal signature in a matter of hours, or will occlude parts of the sky (or the local star)

Hiding Behind a Planet: With transit times measured in weeks and orbital periods measured in minutes, at some point your orbit is going to put whatever object is occluding your observation behind you, and you'll stand out like a bug in front of a mirror.

ENGAGEMENT RANGE: The timing of engagements is set by the force with the greatest ability to change velocity. This results in three basic types of engagements: battles at hyperspace termini, battles at planets, and intercept battles (where one side is trying to prevent the other from reaching a planet or hyperspace terminus). Intercept battles can theoretically happen at tens of kilometers per second of closing velocity; the lack of stealth, and the desire of crews to avoid mutual suicide prevents this handily.

Weapon ranges are limited by the total ability to change velocity for missiles, and by beam spread for beam weapons. Most beam weapons that obey the laws of physics are limited to hundreds or thousands of kilometers, delivering damage in several pulses per second.

SPACE IS BIG; TERRAIN IS SPARSE: The game scale in AV:T is 20km/hex. Planets are represented by an edge of the map (usually the table surface) being impassable to ships of either side, and asteroids are lone clumps of material, one to ten hexes in diameter.

This means that there is no horizon line to hide behind, and thus no need to have "forward scouts" relaying information back to the main force. Likewise, the "clump of asteroids to dodge through while dogfighting" is, alas, a Hollywood invention.

PHYSICS IS THE SAME FOR EVERYONE: Unlike surface naval combat, where aircraft can move roughly 40-50 times faster than their carrier, a small spaceship and a large spaceship have a different dynamic. The smaller ship will mass less, so the same drive mounted on both the smaller ship and the larger ship will produce different amounts of thrust, and the smaller ship will be able to turn faster (due to shorter moment arm effects). The strong distinction between aircraft and surface naval ships doesn't fit space combat, and thus the swarms of fighters swooping and banking as they nimbly dodge the battleship's beams doesn't fit either. (Not to mention the lack of atmosphere in space to swoop and bank against.)

In AV:T, reaction drives are limited by thermodynamics, with thrusts measured in increments of 1/8 of a G. The largest reaction drive used in this product puts out 4.5 terawatts of energy. The total power generation capacity of the human race at the beginning of the 21st century is roughly 12 terawatts. The maximum change in velocity (or delta v, shown as Δv) is set by the fraction of the ship's mass devoted to fuel.

THE ROLE OF THE PLAYER: One of the great divides in the adventure gaming field is split between "competitive" players and role-players. While AV:T is unabashedly a competitive wargame, that should not keep players from thinking about the role they play in commanding a spaceship.

AV:T's default background setting is the Ten Worlds, a story of the human colonization of near stellar space, rife with political machinations, intrigues, fleet actions, tragic errors and transcendent hopes. It is, above all else, a human universe, one where the actions of a few in great duress can alter the course of history, and where the human ideals of compassion, courage, honor and duty reflect all that we aspire to. The political context of the Ten Worlds has moved away from the grinding game of attrition and total war. Limited conflicts for limited political objectives are the norm. It's reasonable to accept a surrender (or surrender yourself) to be ransomed back rather than fight to the last gasp of air.

With this in mind, while it's possible to play AV:T as nothing but a gritty wargame, played to the bloody death for the thrill of beating your opponent with nothing but your tactical cunning, there is the potential for it to be more than that.

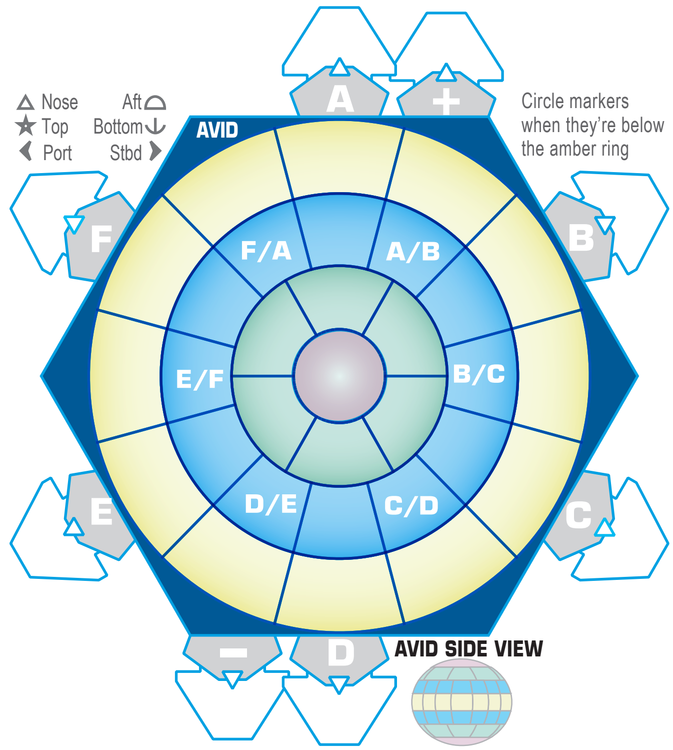
The ship descriptions have detail applicable to campaign games or role-playing games. The historical scenarios flow from a political context, where the results of one battle can influence the next, and (by varying ordnance and consumable stores, or repair status) can be played with incomplete intelligence.

(A2.4) GLOSSARY OF TERMS:

All of these terms are explained in their usage context later in the book.

Crew Rate: The quality of your crew, expressed as a target number to be equalled or exceeded. Ranges from 2+ (Legendary) to 10+ (Hopeless)

AVID: Attitude Vector Information Display; this is the tool used to record changes to a ship's orientation, its current vectors, and changes to vectors. It is also used for shooting bearings, and tracking firing arcs. The key concept of the AVID is that it's a top-down view of a sphere, divided into windows that are color coded in rings (amber, blue, green and purple), showing how far away from the equator you are. The outer part of the AVID is made up of 8 arrows. 6 of them correspond to the A-F map directions, the other two are + (for Up) and — (for Down). Those arrows are called vector arrows, and have a gray part (where changes to vectors are accumulated) and a white part (where numbers for vectors are shown).



There are two different colors of AVID; the Red AVIDs have direction D at the top of the card, while the Blue AVIDs have direction A at the top of the card; if you set Red and Blue so that the arrows on the cards match the directions printed in the center of the map sheet, all directions will "work" for both players of the game without anyone having to rotate their AVID card.

Box Miniatures: A box miniature is the "counter" for your ship in the game. They are printed on card stock with six views of the ship, one printed on each box-side. The top has a triangle pointing to the front and a semi-circle at the stern; think of them as the arrowhead and feathers of the direction you apply thrust in. The bottom of the box miniature has an anchor symbol on it.

Hex Map: The game is played on a map of hexagons, called a hex map. Each hexagon is called a "hex". Ships are placed in hexes, facing either the side of a hex, or a corner, but not on the lines between hexes. The hex map has a rosette in the center, labeled A through F going clockwise. There is a second ring to the rosette, labeled 000 through 330, used for other games Ad Astra publishes. This ring can be ignored for now. Directions Up and Down are shown by + and -.

The maps are *geomorphic*, meaning that when you're about to run off the edge of one map, you can move the second map to that edge and continue the fight.

Stacking Tiles: A ship's position on the map is shown by the hex it's in, while altitude is shown by placing stacking tiles under it, like poker chips.

Tilt Blocks: A ship's orientation on the map is shown with tilt blocks. The box miniature rests in the trough, and can either show a shallow angle (30°) or steep angle (60°). A box miniature that's pointed straight up or down is at a 90° angle.

SSD: Ship Systems Display; the record sheet that shows your ship's current capabilities. Systems can be destroyed on the SSD by marking off their boxes.

Pivots: Any facing change that moves the front of the box miniature around is a pivot, whether it's done up or down or in the plane of the map. If you're used to turning a figure on a map to get it to face where you want, that's a pivot; AV:T allows you to pivot up and down as well.

Rolls: A facing change that rotates the box miniature around its long axis is a roll. Rolling a ship allows you to flip the ship, bringing the other side's weapons to bear.

Bearings: Shooting a bearing is figuring out what part of the sky you see the target in, mapped to one of the windows of the AVID. This generates a Target marker. Target markers get mapped to Firing Arc Diagrams.

A

RALT: The RALT (Range-Angle Lookup Table) is used for shooting bearings. Count out the hexes from you to your target on the bottom of the RALT, then count up by the difference in altitude. The number will be the real range to the target, and the color will match the ring of the AVID it's visible through. The RALT is printed on the back of this rulebook.

Hit Locations: The SSD breaks the ship down into seven hit locations, each with their own table. Which location is hit is determined by a d10 roll.

Vector Movement: Vector movement is like the old video game Asteroids: a ship with a vector of 4 hexes per turn in direction A, 2 hexes per turn in direction B, and one hex per turn in + will move 4 hexes in A, 2 hexes in B and gain 1 altitude each turn. To slow down, you have to rotate 180° and apply thrust in the opposite direction.

Vector: Velocities in AV:T are vectors, and are independent of the ship's facing; they are written as a number of hexes per turn in a map direction.

Thrust: Ships change velocities by applying thrust, which accumulates from turn to turn. Adapting to movement with momentum is the biggest conceptual hurdle for people coming from ground combat games. A ship's current maximum thrust is shown on the right side of the SSD in the Thrust Matrix; damage to the engines marks off boxes on the left, reductions in the mass of the ship (see Fuel Units, below) may shift which row of the thrust matrix is used.

Displacement: If you apply thrust for a whole turn in vector movement, without changing direction, your ship will move a number of hexes in the direction of thrust equal to *half* of the amount you changed your vector by, carrying fractional hexes of movement over to your next turn. This "half movement" is called "displacement". Displacement represents the effect of continuous, rather than instant, acceleration.

Fuel Unit: A fuel unit represents the amount of fuel needed to accelerate the ship by one hex per turn; this is 156.25 meters per second of velocity change. Fuel units are shown on the SSD in the fuel track; each column of fuel units represents one hull space (approximately 25 tons including tank mass) of fuel. The shape of the fuel units changes from circles to squares (and sometimes back and forth multiple times) - each change in shape indicates that the ship's maximum rated thrust has increased due to the ship's mass declining.

Markers: Ships use End of Turn (EoT) markers to indicate where their ships will e at the end of the current turn. They use Future Position (FP) markers to show where the ship will be at the end of the current segment.

Segment: One turn is broken down into eight substeps called segments. Each turn is 128 seconds long, each segment is 16 seconds long.

End of Turn Procedure: At the end of each turn, each ship records how much heat they've stored in their heat sinks, and chooses how many reactors they wish to have active on the next turn. They may also extend radiators.

Reactors: A reactor generates power that refills batteries.

Battery: Power is accumulated in batteries and spent to fire weapons.

Heat Sink: Heat sinks store the waste heat from reactors to be dissipated later, via radiators.

Radiators: The mechanism of the ship used to dissipate waste heat; extending radiators is accepted as a sign of surrender, as the ship cannot sustain combat rated thrusts while they are out.

Kinetic Weapons: Weapons which damage the ship by impacting it; damage is based on the square of the impact velocity and the mass of the impactor.

(A3.0) USING THE GAME

AV:T uses an alphanumeric rule numbering system pioneered by *Star Fleet Battles*. Everything is organized from left to right within the rule number, as described by rule (A3.1) below. While this system can appear daunting at first, in the end, it's an outline, and makes finding things a snap. Ultimately, this rulebook ends up staying in the box when the game is played, and the extensive navigation system is there to make sure that when you DO need to look something up, you can find it quickly

(A3.1) NAVIGATING THE RULEBOOK:

(A3.11) SECTIONS: The game is divided into sections, each designated by a letter. For example, you're now in "section A" which deals with "General Rules and Information." All rule numbers in section A begin with the letter "A", just as this rule (A3.11) does. There are tabs on the edge of each page showing the sections, making it easier to flip through the rulebook and find the right section.
The major rules sections are listed in the Table of Contents; the game also has an Index.

(A3.12) MAIN RULE NUMBER: Between the section letter and the decimal point is a number, specifying a particular rule. Each rule begins with a single column gray box to make it easier to find on the page.

(A3.13) RULES SUBDIVISIONS: After the decimal point, the numbers work differently. Rule (C2.32) is not the 32nd rule under (C2.0), but rather, the second item under the third subtopic of that rule. Rather than indent the outline's hierarchy, progressively lighter shades of grey for the rule numbers themselves indicate levels of rules. The range of rule numbers on a page is listed in the outside corner of the header.

(A3.14) EXAMPLES AND NOTES: Examples of specific rules are shown in the sidebars on the outside edge of each page.

(A3.15) ANNEXES: Some information will be superseded by new releases. An example would be a complete index to the game, or the master ship chart. This information is in Annexes, which are replaced when new supplements come out. Consolidated annexes are sold on the Ad Astra Games web site in electronic format.

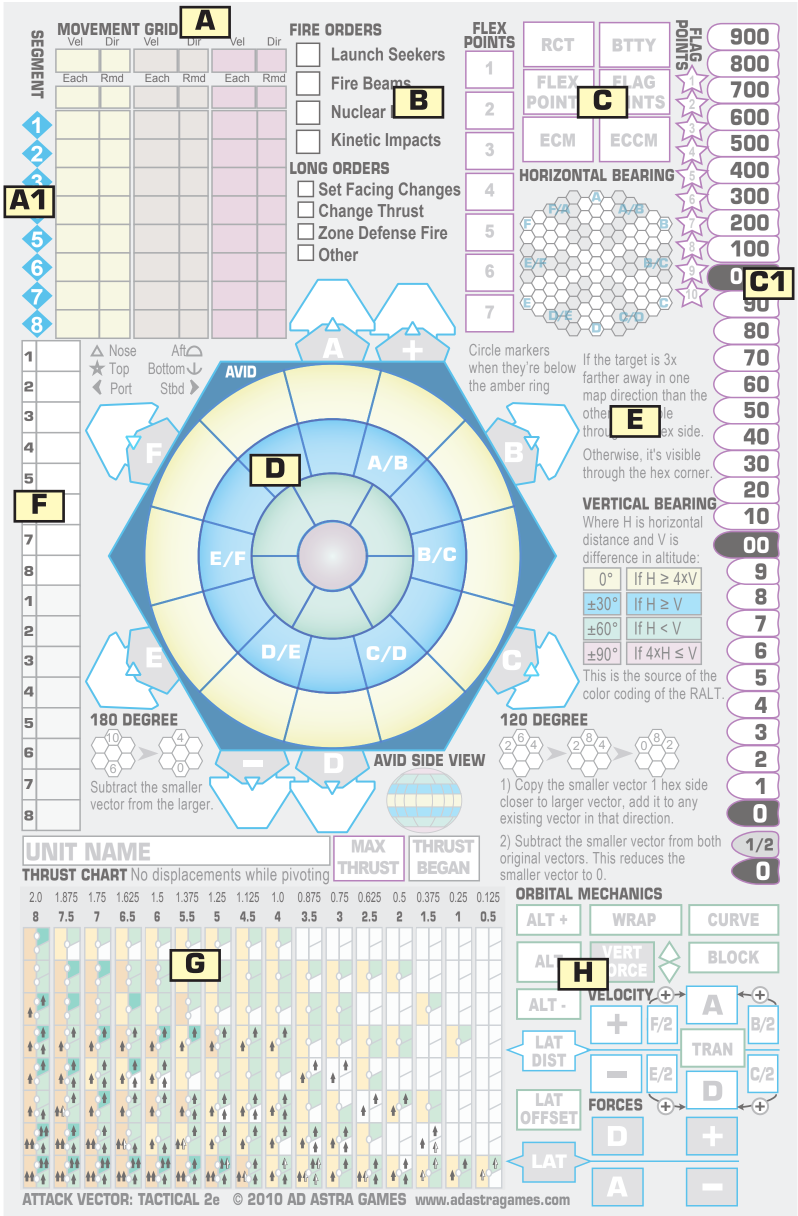
(A3.16) CROSS-REFERENCES AND EXCEPTIONS: Many rules cross-reference other rules in the game; this helps using the rulebook as a reference tool. When exceptions or special cases are listed; they will be clearly marked. If rule A cites an exception to rule B, and rule B does not specifically list the exception from rule A, the exception still applies.

(A3.2) PARTS NEEDED TO PLAY THE GAME:

The following components are required to play AV:T. Many of them can be downloaded and printed out from the Ad Astra Games web site.

- This rulebook, and the ship book. The historical background book has scenarios which are useful for starting a game.
- Photocopies of the Ship Systems Displays (SSDs) needed to play the chosen scenario.
- One box miniature, one future position marker, one end of turn marker, and one Ship Control Card (SCC) for each ship. Each side of the battle should have a copy of the Weapon and Maneuver Reference Card. There should be tilt blocks and stacking tiles.
- At least three 10-sided dice (d10s). More are always useful.
- A hex-map with 0.9" (23mm) or larger hexes. Several vendors supply these, including felt maps sold directly by Ad Astra Games. Two double sided paper maps are included.
- An assortment of pens suitable for using on a laminated surface. Fine tip permanent markers can write on laminate and be erased with rubbing alcohol (or a dry erase marker) and a cloth. Grease pencils or crayons also work.

A



(A4.0) THE SHIP CONTROL CARD

There are eight Ship Control Cards (SCCs) in your boxed game. The ship control card is reproduced here with some of its more important elements highlighted to make seem a little bit less intimidating. Once you know how to play, everything you need for the game is usually on the SCC and the rulebook stays in the box.

(A4.1) FRONT OF THE CARD:

(A4.11) MOVEMENT GRID: At the upper left, marked with an "A" is the Movement Grid. It will be the first thing you check each segment. It has the segment tracker built into it ("A1").

(A4.12) ORDERS CONSOLE: Marked with a B is the place where you check off what orders you're issuing each turn. This will usually be the second thing you check each segment.

(A4.13) SHIP RESOURCES: The area marked "C" is where you record certain ship related resources; this is done when setting up the game, and until the ship is damaged, these values won't change.

(A4.131) Next to that is the power tracking slider, marked "C1".

(A4.14) THE AVID: "D" shows the AVID, which is used for a number of things in the game. It's surrounded by notes. about how things work on it.

(A4.15) BEARING RULES: "E" shows rules for finding a horizontal bearing and a vertical bearing; the small hex grid gives a visual display of the rule underneath it.

DESIGN NOTES: The SCC

The SCC is dense; we used every bit of space we could to make it more useful. It's also ergonomic. The work flow (what you use when) is organized from left to right, top to bottom. Information that you'll need to reference rarely (or will remember after doing a few times) is printed in fainter text so it recedes in the background. Timer tracks are all on the left side of the card, and power tracking is on the right. All thrust related functions are at the bottom of the card.

(A4.16) TIMING TRACK: This track (labeled "F") is used to record weapon delays.

(A4.17) THRUST CHART: This chart (labeled "G") is used to track the effects of thrust from your ship's engine. The space to record your ship's name is integrated into the Thrust Chart.

(A4.18) ORBITAL MECHANICS WORKSHEET: This area (labeled "H") is used to translate velocities in an orbital reference frame into forces that act like accelerations. It is the only part of the front of the SCC that will NOT be used in some way or another in the following tutorial.

(A4.2) BACK OF THE CARD:

(A4.21) HEAT MANAGEMENT: With the exception of the Heat Management Console, the back of the SCC can be ignored for the tutorial.

(A4.3) FURTHER READING:

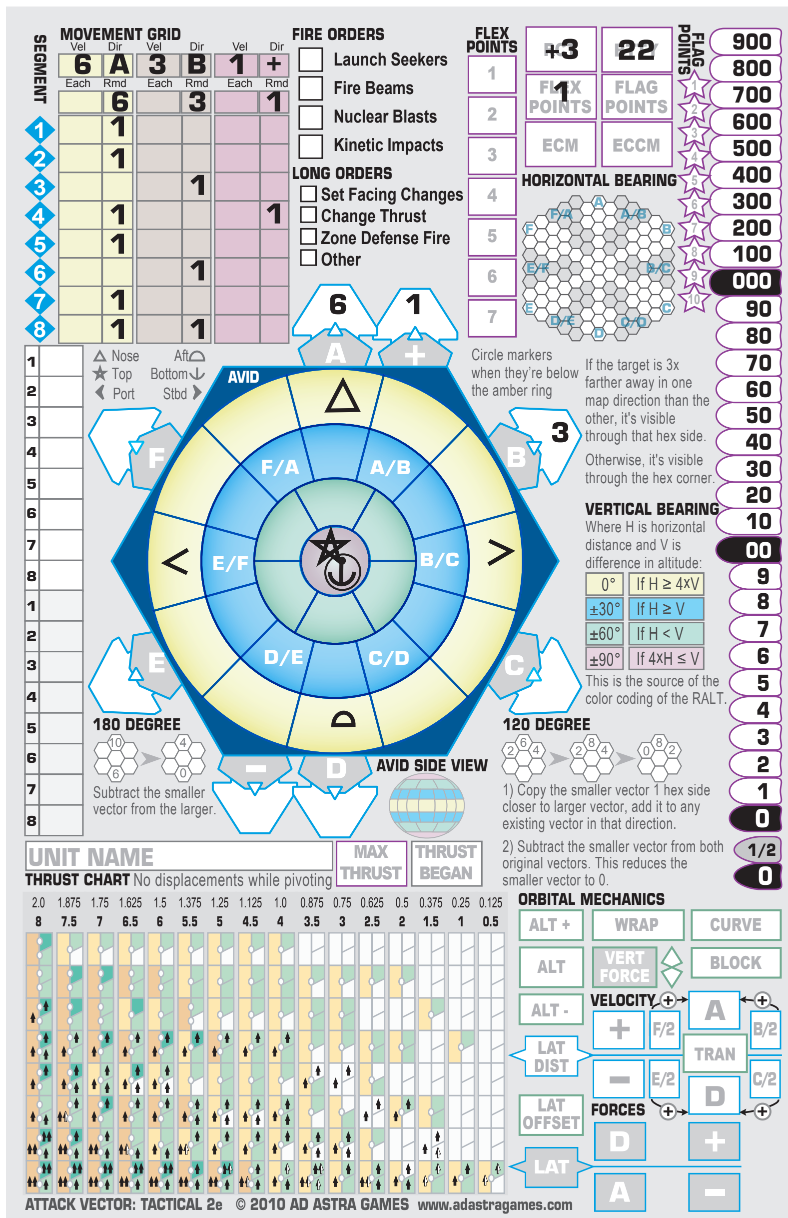
(A4.31) DETAILED EXPLANATION: A more detailed walk through of the SCC is in rule (B4.0).

(A5.0) WALK THROUGH TUTORIAL

AV:T is taught in stages, and by doing, rather than reading. It is very important that you follow this tutorial, rather than just "sit and read it". It covers the highlights of the game and most of the important concepts. To play the tutorial, you will need a copy of the *Rafik Mk. 1* SSD, a movement/weapons reference card, a SCC (preferably one of the blue ones, with direction A on top), a map sheet, some chocolates to use as markers, a set of tilt blocks and some stacking tiles. You will also need some erasable markers or grease pencils.

(A5.1) SAMPLE GAME ONE (DRIFTING, BEARINGS AND WEAPONS FIRE):

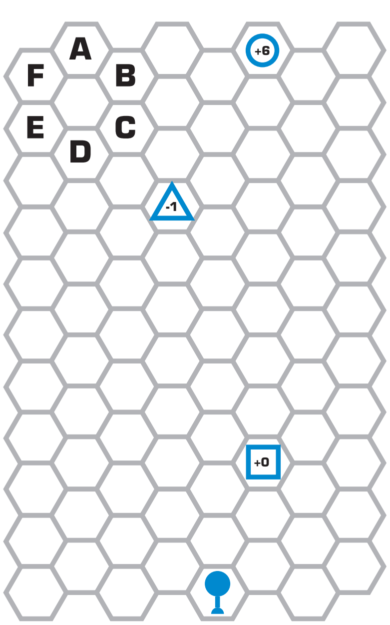
(A5.11) SETUP: Copy the information in the illustration at left onto a blue SCC. Your ship has vectors of 6 hexes per turn in direction A, 3 hexes per turn in direction B, and 1 hex per turn in direction Up (shown with a +). Also, make a photocopy of the *Al-Rafik* SSD, and keep it handy—we'll need it later in the turn.



(A5.111) Place your ship's box miniature on the map. Relative to your ship, place some chocolates at the points on the map shown with a square, a triangle and circle in the map shown at the bottom of the page. The last two chocolates are at –1 and +6 altitude levels respectively. White stacking tiles are one altitude level, light blue are 4, dark blue are 16, and black is a minus sign.

The chocolate at the triangle, 1 level below the map, should have a white tile with a black one on top of it, with the chocolate sitting atop the stack. The chocolate at the circle should have one light blue tile and 2 white tiles (or 6 white tiles) underneath it.

(A5.112) For each additional rule in this section, your ship will drift along its current vector one segment at a time, while various features of the game are highlighted. Read each rule and follow the instructions, then go on to the next rule.



(A5.113) If you fly through the space occupied by a chocolate, you get to eat it. If you fire at a chocolate and do 10 points of damage in one segment, you get to eat it. The ship you're flying has 3 MRLS weapons, so you can use the sample table from earlier in this section, or find it on the weapon reference card.

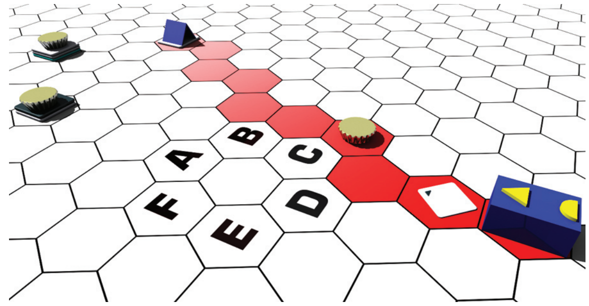
A

DESIGN NOTE: 3D In The Tutorial

Of all the things that throw people, 3-D is the scariest up front, while adding the least mechanical overhead to the game. 3-D *does* requires the most practice to get proficient with. After playing through the first three turns of the tutorial, you should have it down cold.

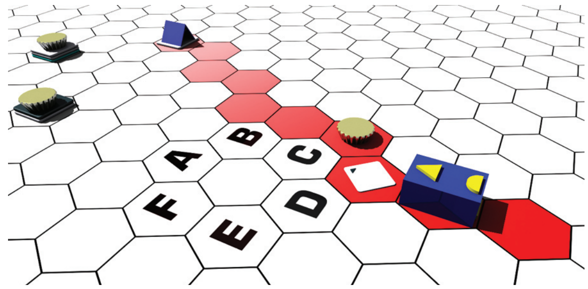
KEY CONCEPT: The AVID

The AVID is a ball that contains your ship; the ball, like the hexes on the map, are fixed - your ship rotates in the ball the same way it can face different sides of the hexes on the hex map. Below this sidebar is a picture of a filled out AVID for this turn, a side view of the AVID showing how many degrees of pitch (vertical angles) correspond to each of the color bands of the AVID, and a rendered picture of Rafik in an AVID in a hex grid, with poles sticking out showing the orientation symbols. The AVID is the most important concept to absorb in playing the game; everything else builds off of it.



(A5.12) SEGMENT 1: On segment 1, your ship moves 1 hex in direction A, shown by the movement grid—look at the row that says "1" on it and read it from left to right. We show this by moving the future position marker at the beginning of the segment, and moving the ship to the future position marker in the middle of the segment. Weapons fire comes after the future position marker has been placed, and before movement.

(A5.121) Place your future position marker one hex away from your ship in direction A, then move your ship to the future position marker.



(A5.13) SEGMENT 2: On segment 2, your ship will move another hex in A. Before moving, there should be a chocolate 2 hexes away from your ship, one hex off in A, one hex off in B. We're going to shoot a bearing on that chocolate, using the AVID (also shown below)

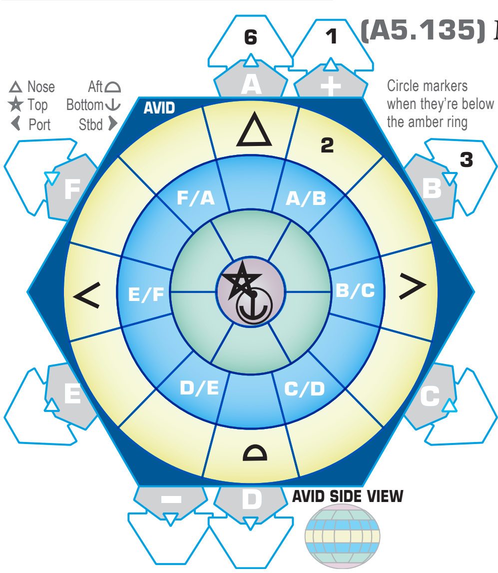
(A5.131) The AVID is a spherical bubble around the ship that fits inside of the hex the ship is in. The AVID is fixed in relation to the map, and not the ship. The view of the AVID on the SCC is from the pole (purple/mauve circle) down. The green, blue and amber rings are subdivided into ***windows***. A side view is shown at right. For this chocolate, which is at the same altitude as our ship, we're only worried about the amber ring. We want to know which amber window we see the chocolate through.

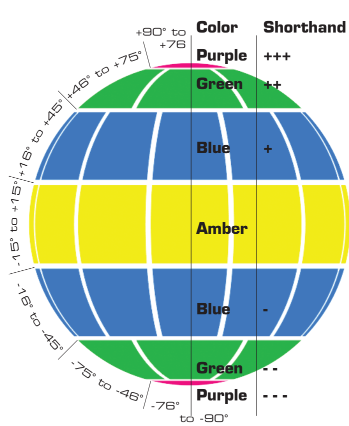
(A5.132) Shooting a bearing means "Where in the sky do I see it?". The end result of shooting a bearing means placing the range to a specific target in an AVID window; the combination of range-in-an-AVID-window is called a 'target marker'.

(A5.133) It's clear from the map that we see the chocolate through the hex corner, between directions A and B at a range of 2. We place the target marker by writing a "2" in the amber window on the A/B hex corner of the AVID, as shown on the AVID at lower left.

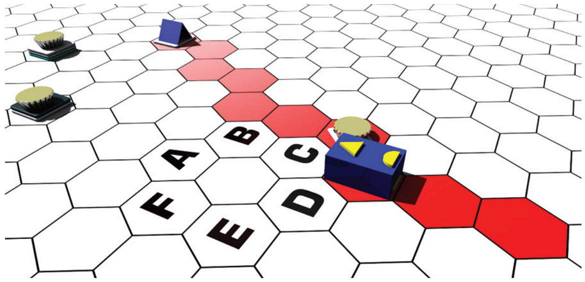
(A5.134) Before moving the ship, try to predict where the target marker for this chocolate will be on the AVID after this segment's movement is complete.

(A5.135) Move the ship to the future position marker.



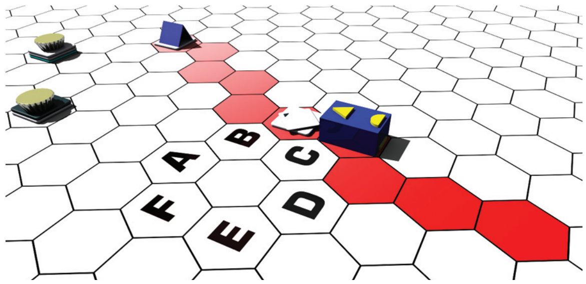


(A5.14) SEGMENT 3: The ship's future position marker will move one hex in direction B. Shoot a bearing and see if it matches your prediction from (A5.133).



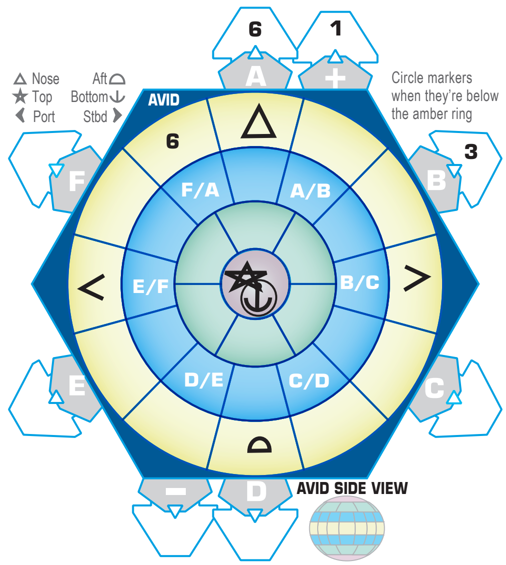
(A5.141) Move the ship to the future position marker and eat the chocolate.

(A5.15) SEGMENT 4: Your ship will move two spaces this segment, one hex in A, and one level up (direction +). Place your future position marker on a white tile, one hex away in direction A.



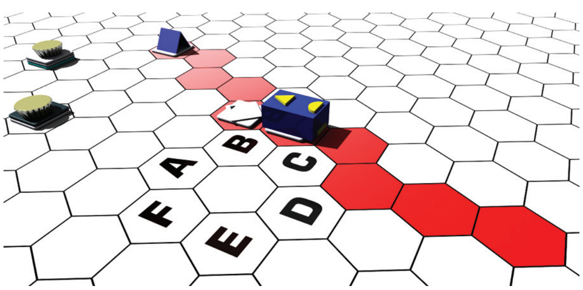
(A5.151) Your course won't take you through the chocolate shown with a triangle.

(A5.152) Let's shoot a bearing and place the target marker for this chocolate on the AVID. From our current position, the chocolate is 4 hexes out in A, and 2 hexes out in F. Would that be visible through the hex corner or the hex edge? If you're unsure, compare the two different distances and use the rule of thumb printed above the AVID—4 is not 3 times as large as 2, therefore it should be in direction F/A.



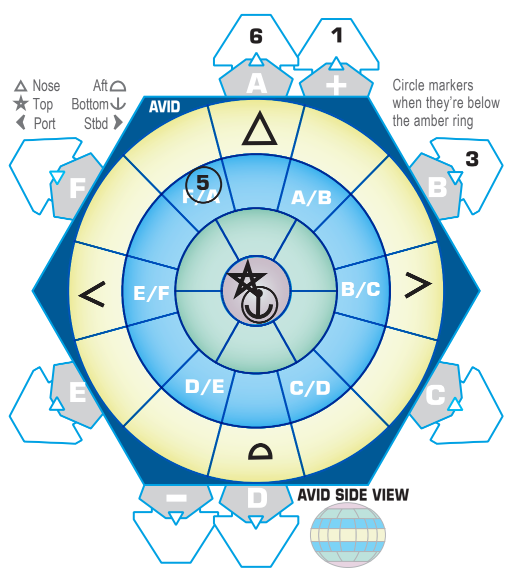
(A5.153) Now look at the chocolate's vertical component: it's one level below us, and 6 hexes away. Using the RALT (Range/Angle Lookup Table) on the back of the book, we'll see that the target is 6 hexes out and has a difference in altitude of 1 hex. This puts it at a range of 6. The color on the RALT is amber, meaning we still see it through the amber ring. We record the target marker in the F/A amber window of the AVID, as shown in the upper AVID illustration in the sidebar to the right.

(A5.154) Move to the future position marker. Your ship should now be one altitude level above the plane of the map.



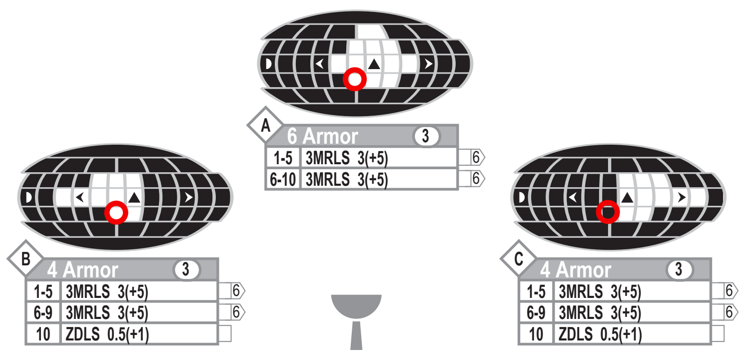
(A5.16) SEGMENT 5: Your future position marker will be 1 hex away in direction A.

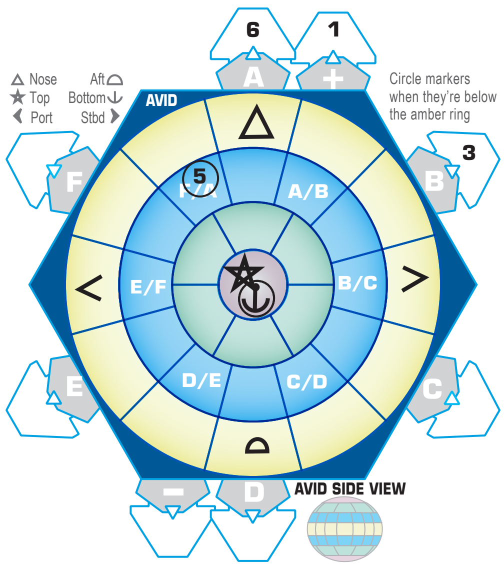
(A5.161) Let's re-shoot that bearing to the triangle-chocolate. It's now 3 hexes away in A, and 2 away in F, which keeps it bearing through direction F/A.



(A5.162) The altitude difference is 2, since we're one level up and it's one level down. The horizontal distance is 5, the vertical distance is 2. Using the RALT, we go out 5 and up 2, and see that the range is 5 in the blue ring. Since it's below us, it'll be in the lower blue ring on the AVID. We write a 5 in the F/A blue ring and circle it to show that it's beneath us, just like the note to the upper right hand of the AVID says. The target marker is shown on the lower AVID on this page.

A





(A5.163) Now it's time to look at the ship. The *Rafik's* forward weapon mounts are printed in the sidebar; the gridded ovals are its firing arc diagrams. The AVID we've been using so far has been a top-down view. Firing arcs are a similar sphere to the AVID—it's divided into 50 windows, but our perspective is different. We're looking out, like a goldfish looking out of a goldfish bowl. Another key difference is that firing arc windows are fixed to our ship, not fixed to the map. The window our Nose is pointed at is shown with a triangle on both our AVID (which is the triangle facing in direction A), and on the firing arc diagram. The top of the ship is always the top slice of the firing arc diagram and the bottom of the ship is always the bottom slice of the diagram. The other orientation symbols (Port, Starboard, Aft and Nose) are all in the center row of the diagram.

(A5.1631) If you're used to traditional hex based games with firing arcs, the middle row of that firing arc diagram—when your ship is level with the map—corresponds to a conventional 12 point clock facing diagram, where the Nose symbol is 12 o'clock, the Port and Starboard symbols are 3 o'clock and 9'oclock, and the Aft symbol is 6 o'clock.

(A5.164) Mapping a target marker (a range to a specific target written in an AVID window) to a specific firing arc window is a separate set of steps from shooting a bearing (placing a target marker on the AVID). Remember, you always place the target marker before mapping it to a firing arc diagram. ***The number one source of confusion in playing this game stems from people who try to do all of the steps in their head. Always write the target marker on the AVID before you try and map it to a firing arc.***

(A5.165) The end result of mapping a target marker to the firing arc is to unambiguously place the target marker into one firing arc diagram window; this window will either be one window away from a firing arc diagram window with an orientation symbol in it, or it will be the firing arc diagram window with an orientation symbol in it. To get there, we have to ask a series of questions, each one of which narrows down the range of possible windows. It's important to do all of these questions, even when the answer is obvious, because the procedure will prevent confusion later on when the ship is at a strange orientation.

(A5.1651) The Top of our ship is the star symbol (the symbols are explained in the key at the upper left of the AVID). How many windows is it from the top of our ship (the star on the AVID) to the target marker? We'd count from the purple window to the green window in direction A (the one in direction F would work just as well), then through the F/A blue window, then to the Amber window in F/A, and then back around to the F/A blue window again. (We've counted out to the Amber ring and then back down in the AVID window). This will tell us which row of the firing arc diagrams we'll see the target through; it's four windows away from the Top of the ship.

(A5.1652) Now that we know which row the target's bearing is in, we count the number of windows the target marker is away from the next closest orientation symbol. In this case, the orientation symbol its closest to is the triangle for the Nose; it's one away and diagonally down and to the left of the ship's Nose. The exact window of the firing arc diagram the target can be seen through is marked in the illustration above with a circle.

I Do This Every Time I Move? Yikes!

Shooting a bearing (figuring out where you see something in the sky on the AVID) and mapping a bearing to a firing arc are two distinct skills—and they are skills that are unlike anything you've ever had to do for a game. The first couple of times you do it, it's going to seem strange. Keep at it. It gets much easier with practice, and it's a place where the game is using a lot of clever math so that you don't have to build a complex 3-D mental model of what's going on. Instead, you shoot bearings to targets of interest only when you think you might shoot at them.

(A5.166) Which of the three forward bearing weapon mounts can shoot at that chocolate? We're looking to fire through a window that is down one diagonal-and-port of the nose. As you can see, only weapon mounts A and B can fire at the chocolate. Now it's time to make a decision—do you shoot now or wait? The weapon you're using (the 3 space MRLS shown earlier) has a point blank range bracket of 0 to 10 hexes, so getting closer won't help; go ahead and shoot. Two 3 space lasers should do the job. Roll a d10 for each one, and cross-reference the result with the range column to get the damage done. All lasers fired at chocolates add their damage together.

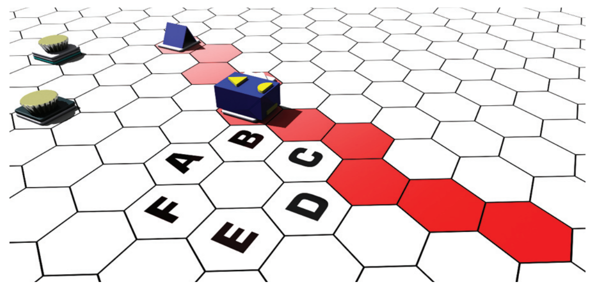
	RANGE > DIE ROLL	0-10	11-13	14-15	16-17	18-22	23-30
3 MRLS 3(+5)	1	5	D7	D5	D5	D3	D1
	2	6	2	D7	D7	D5	D3
	3	6	3	1	1	D7	D5
	4	7	4	2	1	1	D7
	5	7	4	3	2	1	1
	6	7	5	3	2	2	1
	7	7	5	4	3	2	1
	8	7	6	4	3	2	1
	9	7	6	4	3	2	1
	10	7	6	4	3	2	1

(A5.167) In the later sample games, those lasers will have a cycle time and cost power to use, and add some restrictions on how their damage adds together. We won't worry about that for this game, we'll just assume the lasers won't be ready again until next turn.

(A5.168) Now move to your future position marker.

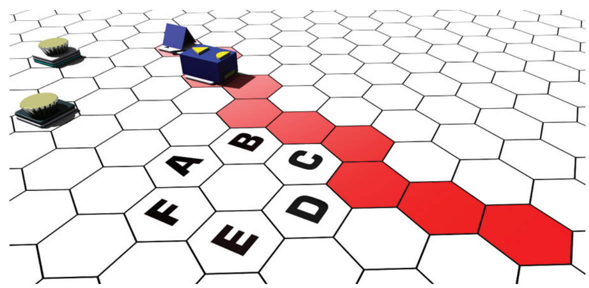
(A5.17) SEGMENT 6: On segment 6, your future position marker will move 1 hex in direction B.

(A5.171) For segment 6, shoot a bearing on the third chocolate, shown with a circle on the initial map. Using the RALT, we look out 6 hexes and go up 5 spaces on the grid; this gives a range of 7, visible through the blue ring of the AVID in direction A. Map the target mark to the firing arc windows like you did earlier. How many weapons can fire at the target?



(A5.18) SEGMENTS 7 AND 8: Continue moving through the remainder of the Movement Grid until you reach the end; the pattern of movements will look like the two illustrations at right on this page. If you feel like putting more chocolates down on the map to experiment with shooting bearings, go right ahead.

(A5.181) To continue your drift for another turn, go back to the top of the Movement Grid and repeat the procedure above. Your ship does not "stop" at the end of the turn, and your vectors won't change until thrust is applied (which is covered in the next sample turn.)



(A5.19) IMPORTANT CONCEPTS: You've learned how to use the Movement Grid, the order of operations in the Sequence of Play, how to shoot a bearing in 3-D, how to map that bearing to a firing arc, and how to read a weapon chart. This is a about 60% of the game right here.

DESIGN NOTE The 2-D Play Experience

AV:T has the 'murderer's row' of scary movement concepts: 12 point facing and firing arcs, segmented movement, vector movement with displacement, fuel tracking and mass reduction and (least familiar of all) 3-D movement. It is possible to turn off 3-D when you play; for some people, a 2-D segmented vector movement game is all they want.

To turn off 3-D, you need to do two things: First, you only use the Amber ring of the AVID, and ignore tilt blocks and stacking tiles. Correspondingly, for shooting bearings, you only care whether the target is visible through the hex side or the hex corner, and you only ever use the middle row of the firing arc diagrams, the row with the Nose, Port, Starboard and Aft symbols.

Second when applying thrust (explained in the next sample turn), you only use the orange cells of the Thrust chart, and the A-F vector arrows of the AVID.

Rules section F assumes 3-D play. Without it, kinetics and seeking weapons become much more powerful, because your ability to evade them through maneuver is greatly reduced. It is strongly advised that you NOT try using kinetics without 3-D.

Orbital Mechanics (rule (C6.0)) requires 3-D play to work at all.

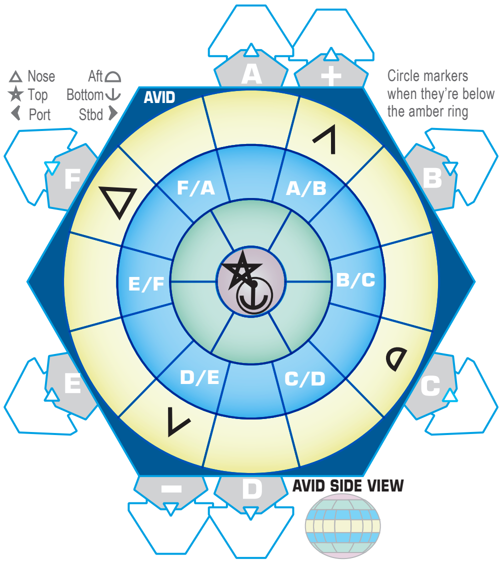
In the end, 3-D requires you to learn some new concepts, but the turn by turn record keeping is minor compared to the play value it adds.

The rest of the tutorial assumes 3-D, as that is the default assumption for the game.

Bearings & Maneuver

One of the ways to identify a skilled AV:T player is that they use the formula references printed in the colored boxes on the SCC to quickly eyeball which window a target is visible through, and then use this from their future position marker to the target's future position marker to see if they're going to get a better shot next segment. Knowing when a target will be visible through a specific window of the AVID allows them to plan their pivots and maneuvers more effectively.

A

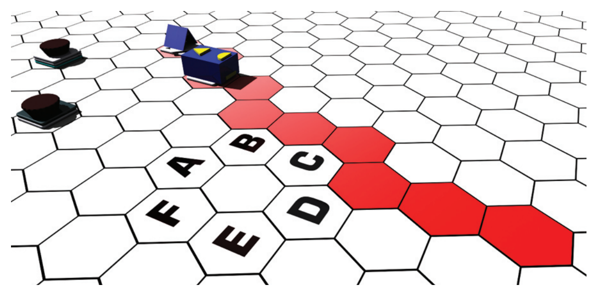


(A5.2) SAMPLE GAME TWO (PIVOTING, THRUST AND VECTORS):

(A5.21) SETUP: Put the chocolates in the same positions as for game one (or shift them a few hexes around for variety). Place the ship in the same place as before, but with no vectors, and facing direction F rather than direction A. This is shown on the AVID in the sidebar here.

(A5.211) You will need the blue SCC and the Weapon and Maneuver Reference Card, as well as a photocopy of the *Rafik Mk. 1* SSD, plus dice, a map, a box mini for a *Rafik*, tilt blocks, stacking tiles, and an erasable marker.

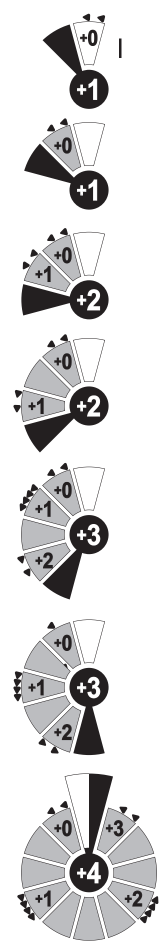
(A5.212) For this sample game, we'll use more of the Sequence of Play. First comes placing the future position marker (the Plotting Step), then comes the Orders Step, where orders are written, followed by Reveal Fire Orders and Reveal Long Orders. Fire Orders are "orders where you shoot something" and Long Orders are "orders that may take longer than a segment to complete". Then comes Resolve Thrust, and finally Movement to the future position marker. We'll also show converting thrust into vectors and how to write out a new movement grid.



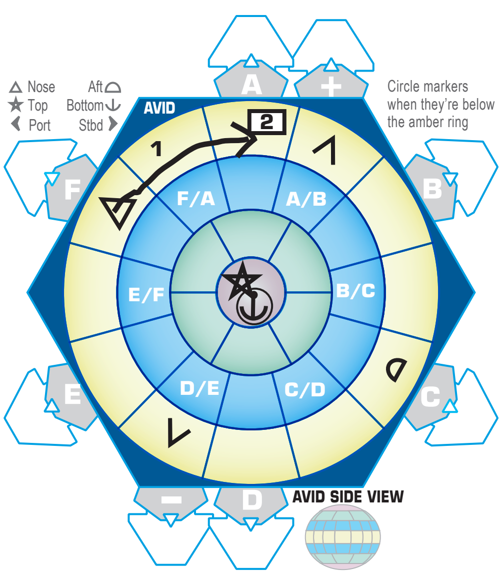
(A5.22) SEGMENT 1: Since your ship has no vectors, its future position marker should rest on top of it.

(A5.221) The first thing to do is pivot the ship's Nose to face direction A. Like the hex map, the AVID is a fixed frame of reference. When your ship pivots, the AVID remains unchanged, and your Nose shifts from one AVID window to another.

(A5.222) On the left hand side of the SSD, there are two columns of "caterpillar charts", one for pivots, one for rolls. The pivot chart for the Rafik is shown to the left.



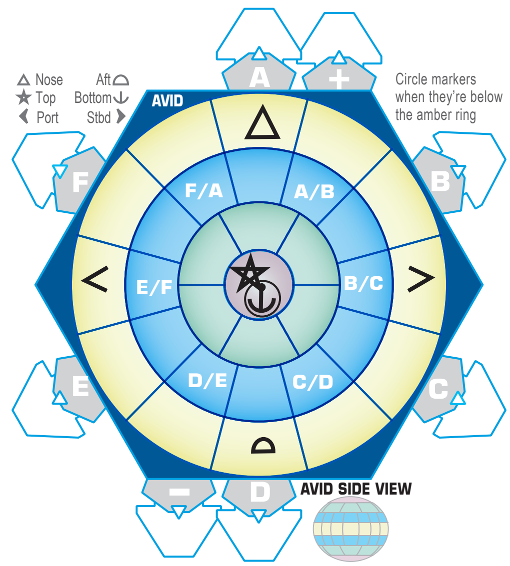
Each wedge on the Caterpillar Chart represents 30° of facing change (ships can face both hex corners or hex edges). The black wedge shows where the pivot stops. Since each wedge is 30°, and we want to change our facing from F to A, that's 60°, or two wedges. The second graphic on the chart has the second wedge blacked in. During the Orders Step, begin pivot/roll is checked and a pivot of 60° is begun. On the AVID, we draw an arrow from the triangle pointing in F (where our ship's Nose is pointing) to the window in A (as shown below).



(A5.223) Pivots take time to complete. To see how long it takes, look at the circle in the center of the chart. In this case, we're doing a 60° degree pivot, so look for the graphic with the second wedge all in black. The number in the circle (in this case, "+1") is the number of segments the pivot will take to complete. The pivot will complete one segment after initiation. Since we're starting the pivot on segment 1, the pivot will be complete during movement on segment 2. We show this by writing the number 2 in the AVID window the pivot ends in, and drawing a box around it.

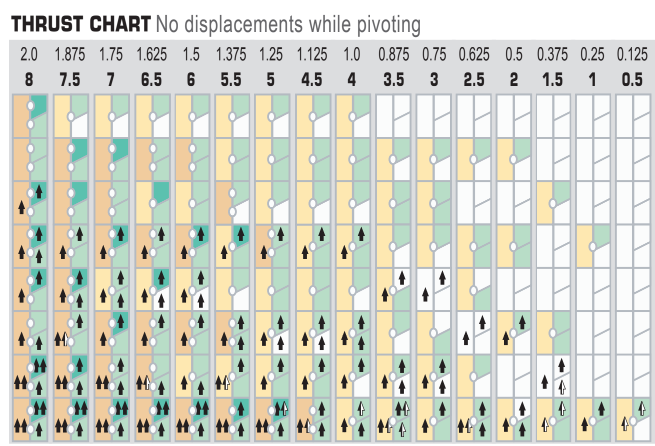
(A5.224) Moving through the Sequence of Play, we haven't engaged thrust yet, so there's no thrust to resolve. The ship has no momentum built up, so the future position marker remains on top of the box miniature. We do have a facing change underway, and the Caterpillar Chart shows a +0 in the 30° wedge of the chart. This means that the ship's Nose will have traversed 30° in the 0^{th} (or current) segment of the pivot; shift the miniature by 30° to face the hex corner between F and A. This is also shown on the AVID on the prior page.

(A5.23) SEGMENT 2: To keep things simple for now, we'll not write any orders on segment 2. With no thrust in progress, there are no thrust effects to record, and the ship is already at its future position marker. We complete the pivot we started earlier, and update the AVID by drawing the triangle in the amber window facing A, and rotating all the other marks so they retain their position on the AVID relative to it, as shown in the AVID at right.



(A5.24) SEGMENT 3: Now that your ship is pointed in the right direction, it's time to turn on your engines. The *Rafik Mk. 1* is capable of a maximum thrust of 5, as shown in Thrust Matrix at the top of the gray box. (We won't be using the second row of the matrix in this tutorial). Set the engine at thrust 5 (1.25 G) and see what happens.

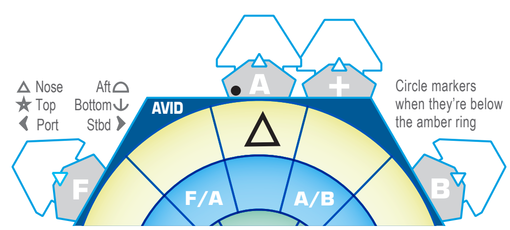
(A5.241) The Thrust Chart contains a lot of information all in one place. Look on the chart until you find the columns for thrust 5. The column on the left is used for "orthogonal" thrust—when the ship is pointed in the plane of the map (or pointed straight up or down). It's shaded in yellow and orange. Between this column and the green column are a set of white ovals; the ovals will become important in a little bit.



(A5.242) During the Plotting Step, the ship's future position marker remains in the hex with the ship. During the Order Step, check "Change Thrust" and write "5" in the box over the Thrust chart. Once a thrust rating has been set, no orders need to be used to keep your ship thrusting—only write a long order when something *changes.*

(A5.243) After orders have been processed, we go to Resolve Thrust and use the 5 column of the Thrust Chart, shown at right. The level of thrust any ship uses is public information.

(A5.2431) A yellow cell generates one *acceleration dot*, which is one hex per turn of velocity change, in the direction the ship's Nose is pointed in. Because this is the first segment under thrust, we check off the first box on the thrust chart (even though it's segment 3 of the turn—whenever we start thrust, we always start from the top of the chart)—then put a dot in the gray (inner) part of the vector arrow in direction A (to show our accumulating vector), like what you see on the AVID here.

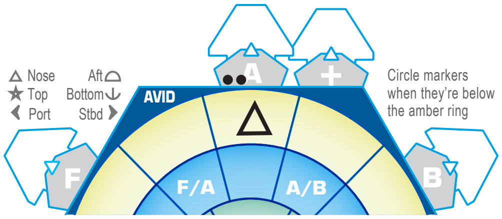


(A5.244) During the Movement Step, our future position marker is still in the hex of the ship, so the ship doesn't move. The ship is not in the midst of a pivot, so its orientation doesn't change. We do, however, check for Thrust Break Conditions as the last item on the Sequence of Play.

(A5.245) Thrust Break Conditions are:

1) Have we gone off the bottom of the Thrust Chart? If yes, we've met a break condition.
2) Have we completed a facing change? If yes, we've met a break condition.
3) Have we set thrust to zero (turned the engine off)? If yes, we've met a break condition.

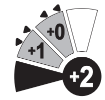
None of those apply, so we haven't met a break condition yet, but should keep them in mind.

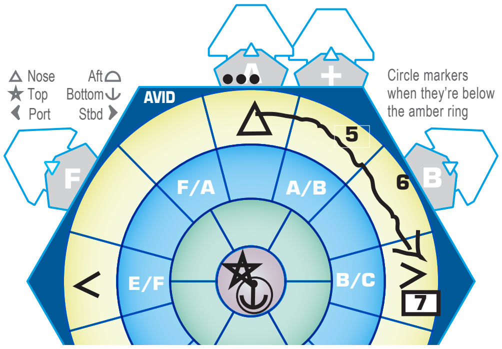


(A5.25) SEGMENT 4: Keeping things simple, the ship's future position marker remains on top of the ship. No changes to orders will be made this segment; the ship will continue to apply thrust, allowing us to skip to Resolve Thrust.

(A5.251) During "record thrust effects", we mark off the next cell in the Thrust Chart, and mark a second dot in direction A. The future position marker still hasn't moved out of the ship's hex.

(A5.26) SEGMENT 5: While the engine is thrusting, we'll begin a pivot to Starboard by 90° (3 AVID windows).



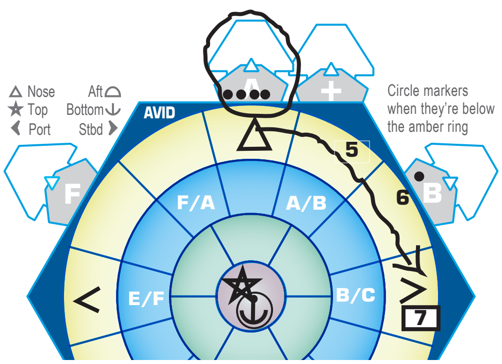


(A5.261) Using the same procedures as segment 1, an arrow is drawn through 3 windows of the AVID. Because the ship is under thrust, the intermediate window information also needs to be recorded. We record the current segment (5) +0 in the A/B amber window, +1 (for a total of 6) in the B amber window, and +2 (for a total of 7) in the B/C amber window, where the pivot stops.

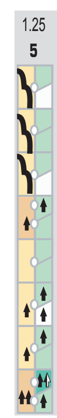
(A5.262) The order things happen on in the Sequence of Play becomes important here. Resolve Thrust happens before Movement, so when we cross off the box on the Thrust Chart, the acceleration dot is recorded in direction A (for 3, total). Then the ship is rotated to face direction A/B. We don't meet any Thrust Break Conditions, so there is no need to resolve vectors yet.

(A5.27) SEGMENT 6: The pivot is still in progress, and the engines are on. How do we resolve the thrust, now that the ship is facing a hex corner?

(A5.271) During the segment, the ship doesn't change any orders. This takes us directly from the Plotting Step to Resolve Thrust, where things get interesting.



(A5.272) First, we choose a bias direction. A bias direction determines where odd acceleration dots go when we accelerate towards a hex corner, and our choices here are directions A and B. We'll put the bias direction as direction A. We show this by circling the vector arrow in direction A. For as long as the ship is facing the hex corner, subsequent acceleration dots alternate between A and B, with odd numbered dots first, third, fifth, etc.) recorded in the bias direction.



(A5.273) The fourth cell down from the top of the Thrust Chart is shaded dark orange this time, the color shows how many acceleration dots the ship accumulates on the current segment. Light orange (as we saw above) is one dot, dark orange is two dots and white is none. (This pattern is repeated in the slanted thrust columns, with light green being one dot, dark green being two.) We place our first dot in A (our fourth in that direction) and the second in B (the first in that direction)

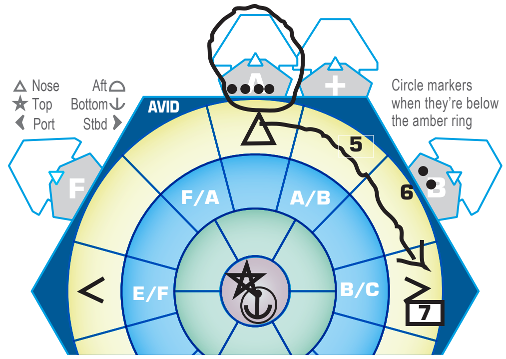
(A5.274) Third, there is a single arrow in the cell on the thrust chart. If the ship were not pivoting, it would *displace* its future position marker one hex in direction A. Two arrows, at the bottom of the chart, results in 2 hexes of displacement. However, displacement doesn't apply when thrusting while also pivoting, we'll ignore it for now.

(A5.275) Because we don't accumulate displacement, the future position marker remains on the ship. At the same point in the Sequence of Play, the ship's Nose moves to face direction B.

(A5.276) This closes segment 6.

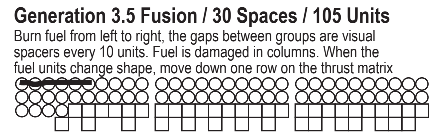
A

(A5.28) SEGMENT 7: As with segment 6, we're still carrying out the implications of our previous orders. No changes in orders are selected, sending us to Resolve Thrust straight away. The ship is facing in direction B, and accumulates one acceleration dot in B (for two total). Once again, because we're in the middle of a pivot, we ignore displacement.

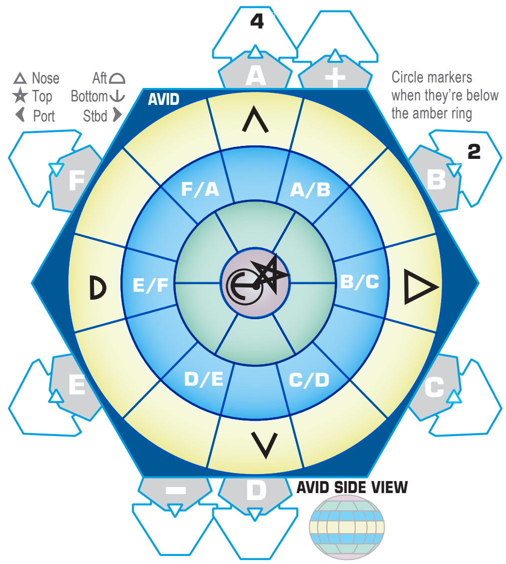


(A5.281) After resolving thrust (gaining one additional dot in direction B), the future position marker remains on our ship. The ship's Nose moves to direction B/C (facing the hex corner between B and C) and the pivot is completed. This is a Thrust Break Condition, and now we resolve vectors.

(A5.282) The first step is finding out how much fuel was spent. Count the dots between the columns of the Thrust Chart. Mark off fuel units on the SSD. Fuel is spent in rows from left to right, with the entire top row marked off before marking fuel on the second row. When fuel tanks are destroyed, mark off columns of fuel units, the left-most first. For now, ignore the fact that fuel units change shape.

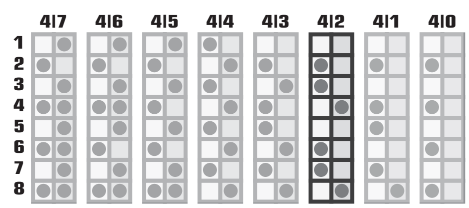


(A5.283) The next step is to count how many acceleration dots have accumulated in each direction of thrust. In this case, 4 in A and 2 in B. Write those numbers down in the white part of the vector arrow, and erase the dots from the gray part, as shown on the AVID at right. Next, we're going to fill in the movement grid.



(A5.284) Record the larger vector (4 in A) in the top row of boxes over the yellow column, and fill out the smaller vector (2 in B) in the tan column.

(A5.285) Next, look at the sheet with the remainder grids, or at the figure below. Since all our movement is in the plane of the map, we're just looking for the pattern of hexes moved, corresponding to 4 hexes in one direction and 2 hexes in another; this is a 4|2 split. First, look down the rows to find 4s, then look across to find the 4|2 split. In the narrow columns of the Movement Grid, copy the pattern of dots from the play aid, turning each dot into a 1. In the remainder grid at right, all of the columns other than the 4|2 split have been made slightly lighter.



(A5.2851) If a given vector is greater than 8, the ship will move 1 hex per segment in that direction, then take what's left over from that to find the pattern on the remainder grid. For example, a vector of 12 in A, and 2 in B would have 1 hex every segment in A in addition to the pattern you just filled in above. It would also take you zipping off the map.

(A5.286) The completed Movement Grid is shown at the bottom left.

(A5.29) SEGMENT 8: The ship's future position marker will be placed one hex away from the ship in direction B. During the Orders Step, change thrust to 0. Remember—if you don't make an order to stop thrusting, you will keep accelerating in whatever direction you're pointing!

(A5.291) Let your ship drift along the vectors as you did on the first sample game. Remember you can pivot to bring weapons into arc.

(A5.292) Experiment with different thrust ratings, thrusting in the plane of the map. Note the differences in displacement and time under thrust between a short duration thrust at a high thrust rating (like the one just completed) and a long duration thrust at a lower thrust rating. Fuel costs are based on the total number of hexes of velocity changed.

(A5.293) When you're comfortable with thrust and pivot in this context, tackle the third sample game.

MOVEMENT GRID

SEGMENT	Vel	Dir	Vel	Dir	Vel	Dir
	4	A	2	B		
	Each	Rmd	Each	Rmd	Each	Rmd
	0	4	0	2		
1						
2		1				
3		1				
4				1		
5						
6		1				
7		1				
8				1		

A

MOVEMENT GRID

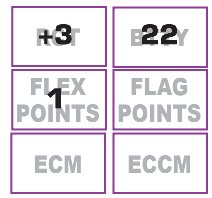
SEGMENT	Vel	Dir	Vel	Dir	Vel	Dir
	4	A	2	B		
	Each	Rmd	Each	Rmd	Each	Rmd
	0	4	0	2		
1						
2		1				
3		1				
4				1		
5						
6		1				
7		1				
8				1		

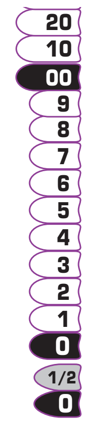
(A5.3) SAMPLE GAME THREE (ORIENTATION & THRUST IN 3-D, POWER AND HEAT MANAGEMENT):

(A5.31) SETUP: Start your ship at the D edge of the map, with vectors of 4 in A, 2 in B, facing in B/C. Place chocolates as shown on the map below; chocolates may be shifted up to 3 spaces in any direction for variety's sake. This is a continuation, of the previous sample game, and the Movement Grid shown at right is identical to the Movement Grid on the prior page.

(A5.311) You will need a photocopy of the *Rafik Mk. 1* SSD (the one from sample game two will do fine), the blue SCC, and the Weapon and Maneuver Reference Card. You will also need the tilt and stacking blocks for your ship, some paperclips, the box mini and a future position marker.

(A5.312) To the right of the AVID, on the edge of the SCC is the Energy Board. Write +3 in the box labeled "RCT", 22 in the box labeled "BTTY" and 1 in the box labeled "Flex Points". These numbers for the *Rafik Mk. 1* are in the data block on the upper right hand corner of the SSD.





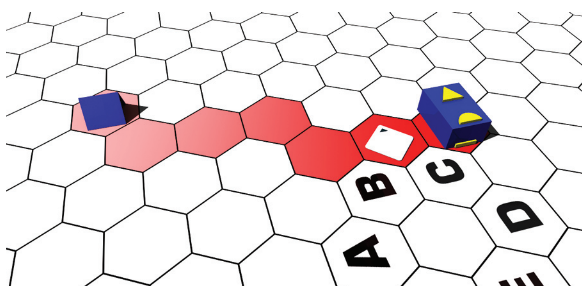
(A5.313) Put paperclips along the Energy Track on the right hand margin of the SCC, one on the "20", one on the "2" and one on the bottom most "0". These paperclips indicate the current amount of energy stored in your ship's batteries.

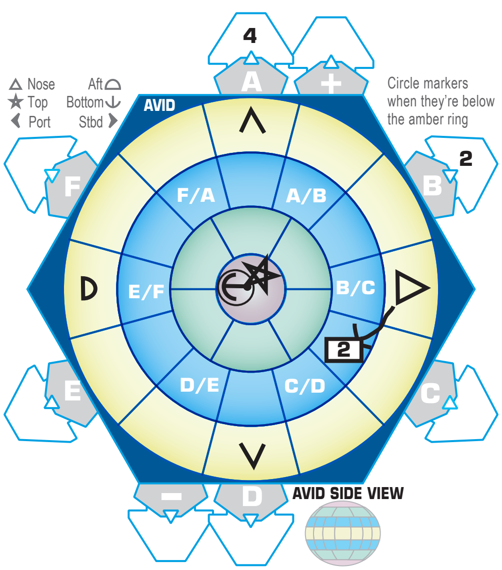
(A5.314) On the back of the SCC, put 15 in the "HEAT SINKS" box, and 3 in the "RCT ON" box. These figures are part of the upper right hand corner data block on the SSD. Put the ship's radiator capacity (1.2) in the Radiator Output box. The ship's radiators are IN, so circle "IN" next to the radiators. (Ships with radiators OUT have surrendered.)

DESIGN NOTE: SCC Color Coding

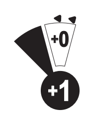
Boxes that have purple borders around them (like the HEAT SINKS box above) are for recording information from the SSD.)

(A5.32) SEGMENT 1: On segment 1, place your future position marker as indicated by the Movement Grid (for this segment it will be in the hex of your ship). The objective on segment 1 is to start moving your ship's Nose from the yellow window in direction B/C, to 30° up (the blue ring) and facing in direction C+.





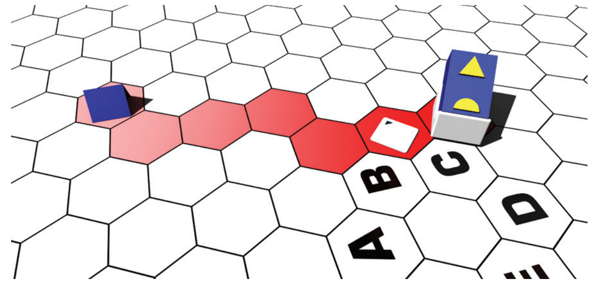
(A5.321) During the Orders Step, check 'Set Facing Change', and draw the shortest path on the AVID from B/C(yellow) to C(blue). Draw a box around the last segment of the pivot.



(A5.322) The ship is not under thrust, so after Process Long Orders, we skip to Movement. The ship's future position marker remains with the miniature, and the facing change's +0 wedge is indicative of where we were facing at the start of the segment, so the ship's Nose does not move.

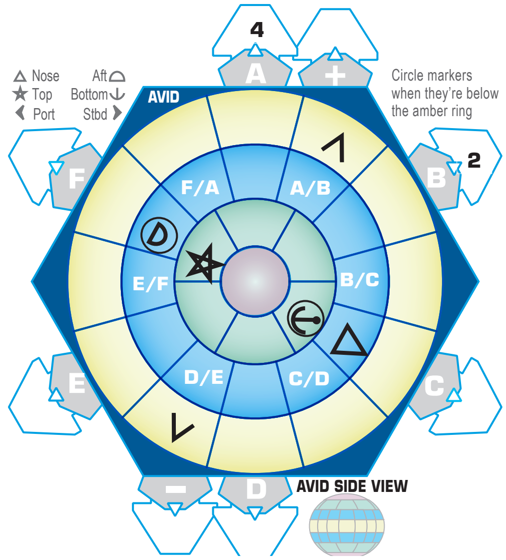
(A5.323) Because there is no thrust, there is no chance of a Thrust Break Condition occurring, so we go to the end of the segment.

(A5.33) SEGMENT 2: Place the ship's future position marker 1 hex away in direction A. There are no orders checked off this segment, and the ship is not under thrust, so we skip to Movement.



(A5.331) The ship moves to its future position marker, and then completes its pivot, with its Nose facing in C(blue). To show the ship's orientation, we update the box miniature and the AVID.

(A5.332) The ship's orientation on the map is shown by turning the box mini so that it faces direction C, then putting the box miniature in the tilt block, so that the Nose points up in the "shallow" angle. The miniature and tilt block system allows everyone at the table to see your ship's orientation, and helps you make more intuitive maneuvers.



(A5.333) The AVID is updated as shown at left. Both the triangle (nose marker) and semicircle (aft marker) are in the blue ring, pointing in C and F respectively, with the Aft indicator in a circle. This shows that your Nose is pointed 30° above the plane of the map, and that your Aft is pointed 30° below the plane of the map. Your Starboard and Port markers should remain in the plane of the map (in the yellow windows), 3 windows clockwise and counterclockwise respectively.

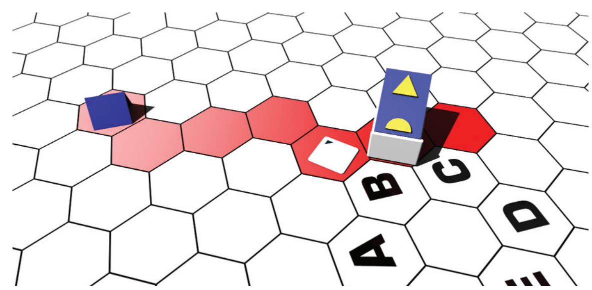
(A5.3331) Since the ship is no longer level with the plane of the map, it's worth noting where our Top and Bottom are on the AVID. Draw a star for the ship's "top" marker, and an anchor symbol for the "bottom" in the positions indicated; circle the Bottom marker. The Top marker will be 3 windows away from both the Nose and Starboard markers, placing it in F(green), and the Bottom marker will be directly opposite the Top (six windows away, counting through F(blue-up), F(amber), F(blue-under), F(green-under), (purple-under) and finally to C(green-under).

(A5.334) As thrust has not been engaged, there are no Thrust Break Conditions, and the segment ends.

Concept: Angled Thrust

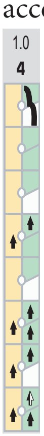
Angled thrust is explained more thoroughly in rule (C2.24)

(A5.34) SEGMENT 3: Now that we're facing in the direction we want, let's engage thrust. The reason we're thrusting in this particular direction will be explained when the burn ends; for now, just follow along to get the procedures down.

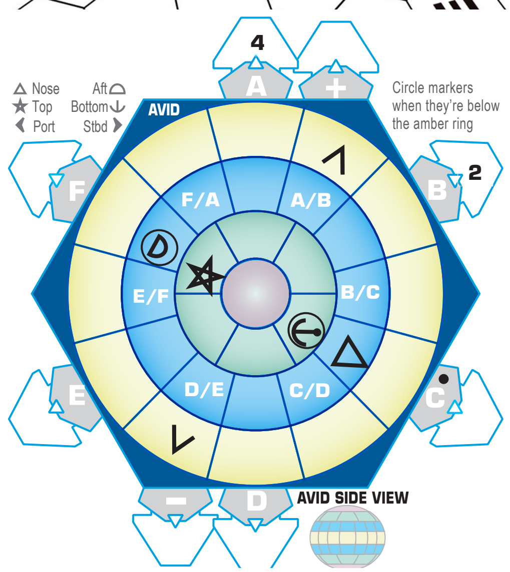


(A5.341) The ship's future position marker is placed one hex away in direction A. During the Orders Step, check "change thrust", and enter 4 in the box.

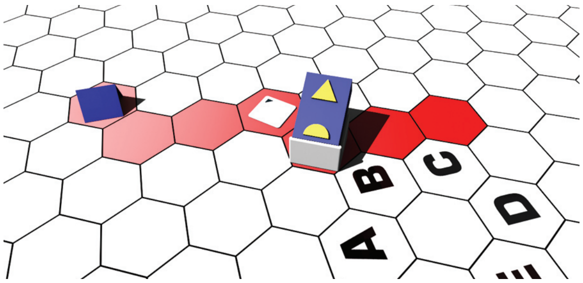
(A5.342) After orders are declared, we record thrust. Looking at the right side of the thrust-4 column, each cell is split with a diagonal line. When thrusting at an angle relative to the plane of the map, some of the acceleration dots will accumulate in the horizontal direction (C in this case, since our Nose is facing in C(blue), and some will accumulate in the vertical direction (+ in this case, since the angle of our Nose is inclined above the plane of the map.)



(A5.343) On the Thrust Chart, the shading and displacement arrows above the diagonal line are for the major thrust component, while the information below the line is for the minor component. So in Resolve Thrust, we accumulate an acceleration dot in direction C, and then check off the entire right hand side of the cell on the thrust 4 column, like the illustration at right. During Movement, we drift to our future position marker (in direction A), and we have no pivots currently outstanding. We meet none of the Thrust Break Conditions, so the segment ends.

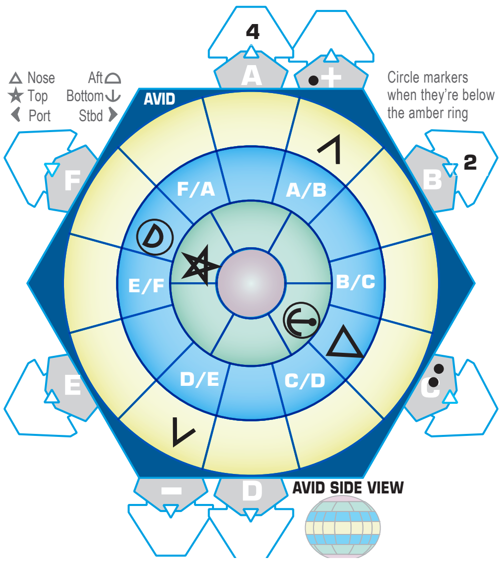


A



(A5.35) SEGMENT 4: Our future position marker shifts one hex in direction B. There are no orders given—we're still thrusting at thrust 4.

(A5.351) During Resolve Thrust, we gain an acceleration dot in C and an acceleration dot in +. We then drift to our future position marker. We're not pivoting just yet, and we have not met any Thrust Break Conditions, so after marking the thrust chart, the segment ends.



(A5.36) SEGMENT 5: Our future position marker remains with our ship this segment, and during the Orders Step, we turn off our engines (set thrust to 0).

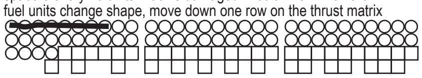
(A5.361) During Resolve Thrust, since the engines are turned off, the ship accumulates no acceleration dots. For Movement, the ship is already at the future position marker, and not in the midst of a pivot. We stopped thrusting this segment, so we meet one of the Thrust Break Conditions.

(A5.362) First, we spend fuel. Looking at the fuel track between the thrust columns, there are two dots, which means we mark off two fuel units from the ship's fuel reserve. Mark fuel expended from left to right on the fuel track of the SSD. Damage to fuel tanks marks off entire columns. Continuing the fuel usage from the turn 2 game, we have burned 8 units of fuel.

(A5.363) Now, we count up acceleration dots. The two dots in C become a vector of 2 in C, and the dot in + becomes a vector of 1 in +. We now have vectors in 3 directions on the horizontal plane of the map, which is confusing. There is a simple way to reduce them.

Generation 3.5 Fusion / 30 Spaces / 105 Units

Burn fuel from left to right, the gaps between groups are visual spacers every 10 units. Fuel is damaged in columns. When the fuel units change shape, move down one row on the thrust matrix



(A5.364) Check to see if any vectors are directly opposite each other. For example, a vector in A and a vector in D. (There aren't in this example, but it's the first thing to check.) Second, check to see if there are any vectors that are 120° (four AVID windows) apart. An example of this is the vector of 2 in C and 4 in A. Using the rule of thumb printed below and to the right of the AVID on the SCC, we copy the smaller vector 60° closer to the larger one. This gives us vectors of 4 in A, 4 in B and 2 in C. Then we subtract the value of the smaller vector from both of the vectors 120° apart, which leaves us with a vector of 2 in A, and 4 in B. We need to redo the Movement Grid with the new vectors. In this case it's easy—because the larger vector is always put in the yellow column, we can keep the grid we currently have and just change the directions at the top of each column.

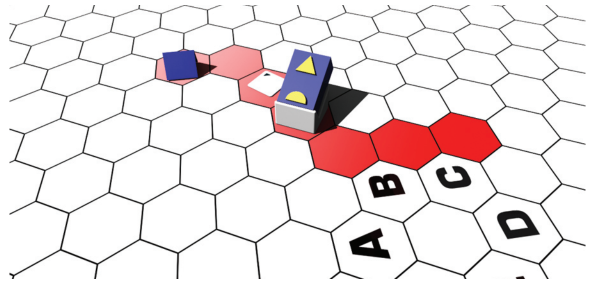
MOVEMENT GRID

SEGMENT	Vel	Dir	Vel	Dir	Vel	Dir
	4	B	2	A	1	+
	Each	Rmd	Each	Rmd	Each	Rmd
	0	4	0	2	0	1
1						1
2		1				
3		1				
4				1		
5						
6		1				
7		1				
8				1		

(A5.365) This leaves us with that vector of 1 in + to resolve. Vertical vectors always use the pink column. 8 goes into 1 0 times with a remainder of 1. Now we need to look at the Remainder Grid on the Maneuver Reference Card again. Add up the combined remainders of the horizontal components (4 and 2) resulting in 6. Treat this as the first part of a remainder split and the 1 as the second part, and look up that split on the Vertical Remainder Grids. Record the dots in the pink column of the Remainder Grid onto the pink column of your Movement Grid. Don't worry too much about memorizing these steps—they're printed below the Remainder Grid as a memory aid. Your combined vector is a 4|2|1 split, shown on the Movement Grid on this page.

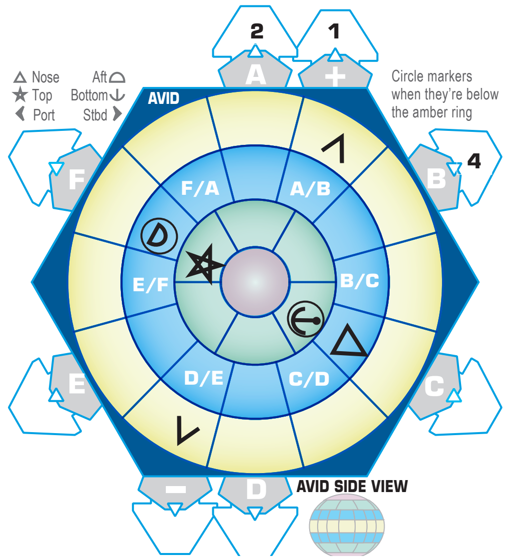
(A5.366) This burn was selected to show the second worst possible case for vector consolidation. Had the ship been thrusting in C/D+, it would have accumulated an acceleration point in C, D, and +. Can you work out the vector split from that case? Can you figure out where you want to be facing (and applying thrust) to reach the chocolates on the map?

(A5.37) SEGMENT 6: Place your future position marker, and decide the pivot you need to get a weapon into arc with the *Rafik Mk. 1*. The weapon mounts have broad arcs on this class of ship, so it shouldn't be too hard to find something in arc.



1	
2	
3	
4	
5	
6	
7	
8	
1	
2	
3	A1
4	
5	
6	
7	
8	

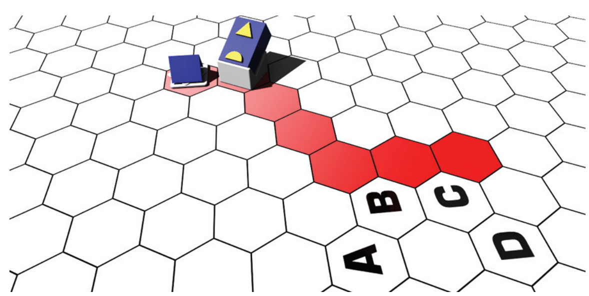
(A5.371) If you have a laser in arc, it's time to learn a little more about how laser weapon codes work. The weapon mount for a 3 space laser on a *Rafik* is 3 MRLS 3(+5). The first 3 is the number of spaces the weapon takes up; lasers range from size 2 to size 8 (and get larger in later products). The MRLS means it's a medium range laser. These tell you which weapon table to reference on the weapons sheet. The second 3 is how many points of power (from batteries) it takes to fire, and the +5 in parentheses is the cooling time (how many segments it takes between firings). The weapon delays are tracked by writing the weapon mount and row (the first laser in mount A would be A1) in the shaded boxes on the timer track on the left side of the SCC. If we fire two lasers on segment 6, they will be available to fire again in 5 segments, or segment 11, which is segment 3 of next turn.



(A5.372) If you fire two lasers on this segment (the minimum needed to kill the chocolate), you'll spend 6 battery power. Record this by sliding the paper clips on the Energy Track from 20 and 2 to 10 and 6. This shows you'll have 16 energy left in the batteries.

(A5.373) The ship is not thrusting, so we skip Resolve Thrust. The ship does drift to its future position marker, and (if pivoting) continues through its pivot. Check for Thrust Break Conditions (and resolve them if necessary). Now we add another step to our Sequence of Play—Regeneration. The ship has two reactors turned on, each generating 1.5 power. This generates 3 points of power every segment, which gets stored in batteries. Slide the paperclips up to 10 and 9 from 10 and 6. The maximum power your ship can store is equal to your battery total.

(A5.38) SEGMENTS 7 AND 8: Drift along your vectors on segments 7 and 8, and shoot chocolates as the whimsy strikes you; remember to slide your paper clips for power tracking and timing tracks as you do this. Practice going through all the steps; by now, they should nearly be automatic.



(A5.39) POST-SEGMENT 8: At the end of the turn, there is a minor amount of record keeping. The first is accumulating heat points; reactors generate heat. Each turn that a reactor is active, it generates heat points equal to the full power generation of the reactor, measured in power per segment; thus, the Rafik generating 3 power per segment generates 3 heat at the end of the turn. Fractional heat points are always rounded up before heat dissipation. Heat is stored in heat sinks. Record these in the "current heat stored" box on the back of the SCC. You can choose to turn off reactors (generating no heat or energy), or turn them back on.

(A5.391) Extending and retracting radiators takes a full turn (for either operation). Once extended, each radiator dumps 0.4 heat points per turn, and any fractional heat points remaining after dissipation are rounded up when radiators are retracted. Radiators are vulnerable to weapons fire; a ship can only apply thrust when the radiators are retracted. Extending radiators is a surrender signal.

(A5.392) Reactor and radiator status can only be changed between segment 8 and segment 1; the status of the systems chosen remains that way for the duration of the next turn.

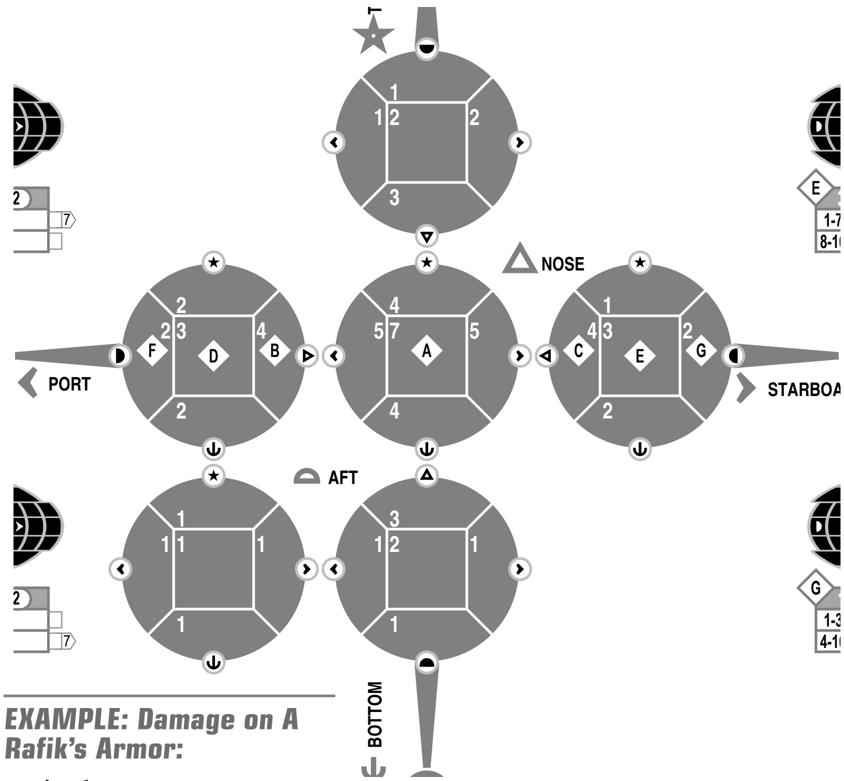
A

(A5.4) SAMPLE GAME FOUR (WEAPON FIRE AND DAMAGE ALLOCATION):

(A5.41) DESCRIPTION: Damage allocation breaks down into three steps, once the damage to the target is known. The first is finding what AVID window the incoming attack comes through. The second is mapping from the AVID to a surface region of the ship. The last is destroying systems as the damage punches through the ship.

(A5.42) INCOMING DAMAGE BEARING: All bearings are reciprocal. If the firing ship sees the target in AVID window D (blue, lower), the target will see the firing ship in window A (blue, upper). The targeted ship marks that window with an "I" for "incoming damage." Making things easier, the red ship control cards have direction D printed at the top of them, allowing red and blue SCCs to be set to set up on opposite sides of the map and share a common reference frame.

(A5.43) MAPPING TO A SURFACE REGION: Looking at the armor diagram of a *Rafik*, each of the six faces of the ship is divided into 5 regions. Armor regions are shown from the inside of the ship looking out. The center region of the Nose group corresponds to damage coming in through the Nose window of the AVID. If damage comes in from one window to the right of the nose, it hits the armor region one to the right of the nose. Armor on the Aft of the ship is viewed from the outside, to keep "right is starboard" as a useful orientation rule. Likewise, damage coming in through the Port marker window on the AVID would come into the center region of the Port side armor, and so on for Top, Bottom, Starboard and Aft.



(A5.431) If the AVID window maps to a boundary between two regions, the attacking player chooses the region hit.

(A5.432) On the *Wasp*, the armor regions "one off the nose" are the foremost regions on each of the four sides of the ship. The "one off the aft" regions are the armor regions at the end of each of the four sides of the ship.

(A5.433) While not used in the tutorial, rule (D4.1) has rules for shifting your aim point to different regions, including the mast and engine, and targeting weapon mounts.

(A5.434) Some armor regions have diamonds with letters on them, these armor regions have weapon mounts on the skin

EXAMPLE: Damage on A Rafik's Armor:

We're doing 36 points to a Rafik. The Rafik is facing direction A/B, and the damage is coming in from direction A(blue, upper). Looking at the armor diagram, damage is coming in one window diagonally above and to the left of the Nose window, mapping to the border between two regions, one ot the Port (with 5 armor) and one above (with 4 armor) the Nose. The attacker chooses to hit the region with 4 armor, getting a soak roll of 1, meaning 31 damage points reach the interior of the ship.

(A5.44) ARMOR AND SOAK ROLLS: Each surface region has a white number in a gray background; this is the minimum amount of damage the armor blocks. Additional damage will be blocked with a soak roll, which is generated by rollin two d10s, subtracting the smaller from the larger) from the damage done. Skin armor of 0 does not generate a soak roll.

(A5.441) Soak rolls that come out to 0 do a Structural Integrity hit to the ship; the SI track is the second row of boxes in the gray area on the right hand side of the sheet, right underneath the Thrust Matrix.

(A5.442) If the armor on the skin is equal to or greater than the amount of damage done, no soak roll is needed.