

Modelling Rocket Motor Propellant Grain Burn-Back

Bsc. Prosjektplan

Jon Anders Sylvarnes

Martin Holltrø Spongsveen

Mål og rammer

Bakgrunn

Nammo Gruppen er en teknologidreven romfart- og forsvarsvirksomhet som spesialiserer seg på høy ytelses løsninger. I porteføljen deres finnes blant annet rakettkastere, militær- og sportsammunisjon, rakettmotorer for militær- og romfartsbruk og miljøvennlige demilitariserings tjenester¹.

De ønsket seg en simulator for forbrenningen i sine fastoffraketter. Versjonen Nammo bruker i dag er en utdatert 2D versjon som er veldig upraktisk og vanskelig å bruke. De kom derfor til NTNU på Gjøvik med en bachelor oppgave, for en mer høyteknologisk simulator.

Formen på drivstoffet i fastoffraketter bestemmer hvor stor brennflaten er, som bestemmer kraften som blir produsert. Ved å kutte former i brennstoffet kan en få varierende mengder kraft under forbrenningen. Vanligvis vil en ha mye kraft ved utskyting, for så å få mindre kraft for å opprettholde marsjhastighet.

Prosjektmål

Effektmål

Nammo ønsker med sin løsning følgende:

- Økt effektivitet ved utvikling av rakettmotorer gjennom redusert tidsbruk i modelleringsfasen.

Resultatmål

Programvaren vil ved ferdigstillelse av løsningen levere:

- Windows
- Programvaren skal kunne levere en tabell med beregninger av effekt over brenntid.
- 3D modell av drivstoffet og graf av forbrenning
- 0.5% nøyaktighet

Læringsmål

- Lære python og mange av de vanligste bibliotekene der, samt mer spesialiserte biblioteker rettet mot 3d modellering.
- Bedre innsikt i systemplanleggingsprosessen og spesifikt gjennomføring av denne planen, med hvilke avvik og endringer som vil bli nødvendige.
- Scrum systemutviklingsmodellen, og gjennom den bli flinkere å estimere tid- og resurssbruk i forbindelse med programvareutvikling.
- 3d modellering med implisitte flater, med de matematiske og programmatisk utfordringer det krever.
- Lære Latex formatering for å skape rene, profesjonelle og strukturerte dokumenter.

Rammer

I og med at dette er en bacheloroppgave, er målet å være ferdig med programvaren innen 12 uker, for å sette av tid til ferdigstilling av oppgaven. Den absolutte tidsrammen er 14 uker i tilfelle uforutsette hindringer.

Omfang

Fagområde

Når Nammo skal utvikle en ny fastoffrakett starter de med en ønsket effektkurve, og må så finne en form for drivstoffet som gir den ønskede kurven. Dette gjøres gjennom eksperimentering, der de lager en modell for formen, beregner effektkurven og sammenligner den med den ønskelige kurven. De verktøyene de bruker til dette i dag er gamle, upraktiske og gir begrenset outputdata. Det er noen som har prøvd å gi ut noen alternativer, men ingen av disse er praktiske å bruke. Vi har da valgt å ta for oss en løsning som skal løse dette problemet og gi Nammo et verktøy de virkelig trenger.

Avgrensning

Under samtaler med Nammo kom det fram at den optimale løsningen hadde vært å lage en programvare som tar inn en effektkurve, som gir ut en ferdig formet brennstoffmodell. En slik programvare er noe som alle innen rakettforsknings bransjen kunnet tenke seg. Det oppstår da et problem ettersom en graf kan gi ut mange forskjellige former som passer grafen. Det er vanskelig å få en maskin til å

finne den beste løsningen, og skille ut løsninger som ikke kan produseres. Dette er noe som blir alt for komplekst for en bacheloroppgave. Vi vil lage en programvare der de kan bestemme utformingen på brennstoffet og umiddelbart se hvordan det vil påvirke grafen.

I de aller fleste tilfeller bruker bare Nammo variasjoner av stjerneutformingen for å styre effekten, siden denne formen kan gi veldig mange variasjoner avhengig av antall tagger, lengde og bredde på tagger og så videre. På grunn av dette vil vi avgrense formene en kan velge til rette sylindere og stjerner, med tilhørende parametere.

Oppgavebeskrivelse

Vi skal levere et produkt som lar brukeren definere en 3d modell av drivstoffet og se hvordan effektkurven vil være umiddelbart, det skal også gi verdiene av kurven i tabellform.

Hovedfunksjonalitetene vil da være:

- Et 3d modelleringsverktøy
- Beregning av effektutvikling over tid
- Presentasjon i graf- og tabellform

3D Modell:

Brukeren skal kunne bygge opp en komplett drivstofftank, med komplett brennstoffprofil, basert på forhåndsdefinerte segmenter med gitte parametere for modifisering.

Segmentene med parametere er:

- Form og lengde på tank
 - Brukeren skal kunne velge lengde og diameter på tanken og hvor avrundet toppen skal være, bunnen trenger ingen modifisering da den er alltid flat.
- Lengde på hvert segment i raketten.
 - Segment vil vanligvis være adskilt av slottet, men kan også være definert av en endring i utformingen (sylinder glir over i stjerne)
- Ytre og indre radius på sylinderform
 - Den ytre radiusen er der brennstoffet møter innsiden av tanken.
 - Den indre radiusen er hvor brennstoffet møter brennkammeret.
- Utformingen av stjerneform
 - Hver arm er parallell og enden er avrundet
 - Brukeren kan endre på bredden og lengden på hver arm og antall armer.

- Vinkelen mellom hver arm er lik, så antallet armer og størrelsen på hver vil implisitt bestemme vinkelen
- Radius, tykkelse og lengde på slot
 - Det skal være muligheter for å bestemme radiusen på en “slot”, dette er fordi under produksjon vil de kuttes ut med et verktøy som skjærer i en bueform.
 - Brukeren skal kunne bestemme tykkelse og lengden på en “slot” for å få ønsket resultat.
- Tykkelse på isolasjon
 - På stedene det har blitt lagt inn “slot” er det behov for ekstra tykk isolasjon. Den ekstra isolasjonen vil bli lagt på innsiden av tanken som gjør at det blir mindre plass til brennstoff i tanken.

Beregning

En av de viktigste delene av programvaren er beregningen av forbrenningen i tanken, da hvis denne er unøyaktig er programvaren nær verdiløs. Vi ser for oss å utvikle en algoritme for hvert segment brukeren kan legge inn i tanken, men dette er ikke noe vi har erfaring innen og vi er åpne for endringer i fremgangsmåten når vi undersøker temaet nærmere.

Presentasjon

- **Numerisk output:**

Nammo vil ha muligheten til å få resultatene levert i et numerisk format, der de får effekten over tid presentert i en tabell. Dette skal kunne brukes som input til andre programmer, så det nøyaktige formatet må bestemmes etter nærmere samtaler med Nammo.

- **Graf:**

Når brukeren har laget sin versjon av brennstoffet skal resultatene vises i en graf umiddelbart. Grafen vil kun vises så lenge modellen er komplett, før dette og om brukeren fjerner et segment vil grafen ikke vises. Så lenge modellen er komplett skal alle endringer oppdatere grafen med en gang. Algoritmen til grafen skal vises til brukeren i høyre hjørne.

Prosjektorganisering

Ansvarsforhold og roller

Vi har valgt å ikke ha noen prosjektleder ettersom vår gruppe er veldig liten. I en gruppe med 3 eller 4 elever kan det være behov for en prosjektleder, men ettersom vi er en gruppe 2 elever synes vi det er unødvendig å sette en av oss som prosjektleder.

Vi mener å ha funnet en effektiv måte å løse uenigheter, gjennom vetorett, se [vedlegg A](#) for nærmere informasjon. Vi velger å jobbe tett sammen ettersom dette er et kryssfaglig prosjekt og vi har forskjellige ferdigheter og styrker. Ved å jobbe tett sammen kan vi lære av hverandre og raskere oppdage feil og misforståelser, som fører til mer effektiv jobbing og et helhetlig resultat.

De følgende rollene er angitt i prosjektet:

Produkteier:

Nammo Raufoss

Kontaktpersoner:

Erland Ørbekk

Nils Kubberud

Veileder:

Ivar Farup

Kommunikasjonsansvarlig:

Martin Spongsveen

Grupperegler og rutiner

For grupperegler, se vedlegg A

Rutiner:

- Alle commits til repoitoriet skal kun inneholde fungerende og testet kode
- All kode skal være kommentert
- Gruppen skal møtes kl 10:00 på hverdager
- Alle gruppemedlemmer skal logge arbeidstimer daglig

Planlegging, oppfølging og rapportering

Valg av SU-modell

Oppdragsgiver uttrykte et ønske om tett oppfølging og innsikt underveis i prosjektet og god dokumentasjon for fremtidig vedlikeholdsarbeid. Det var også mulighet for endring eller tillegg av funksjoner i programvare, så en agil tilnærming var naturlig.

På grunn av kravet til dokumentasjon tenkte vi raskt på Scrum som en god løsning og gruppen var allerede godt kjent med denne modellen. Vi undersøkte en del andre utviklingsmodeller, der Kanban og RUP kom frem som gode alternativ. Kanban er litt for løst strukturert for såpass uerfarne utviklere, siden det ikke er noen tidskrav for når hver funksjonalitet skal være ferdig. RUP er svært komplekst og ville ført til mye lengre planleggingstid om vi skal implementere den best mulig, siden vi ikke har noen erfaring med denne modellen.

Under undersøkelsene kom vi over flere element fra andre utviklingsmodeller som vil være fordelaktige å implementere i vårt prosjekt. FDD har gode rutiner for å identifisere funksjonaliteter som legges til i backlogen² og XP har mange gode verktøy for å standardisere koden og vi vil også implementere pairprograming i størst mulig grad. Hvis en av oss ikke er i stand til å møte, eller en part jobber mer enn planlagt, vil vi gå gjennom endringene så fort vi har mulighet til å møte.

I implementasjonen vår av scrum vil Erland Ørbekk og Nils Kubberud fra Nammo ha rollen som Product Owners og Martin vil ha rollen som Scrum Master. Fordi vi ønsker å være selvorganiserende og lære mest mulig om estimering, vil Scrum Master ta på seg en del av ansvaret til Product Owners. Dersom Product Owners har noe spesifikt som de ønsker å prioritere vil det bli prioritert, men de har uttrykt et ønske om en mer veiledende enn bestemmende rolle i prosjektet.

Vi vil følge en to ukers sprint syklus, men vi forventer noen feilberegninger underveis, så dersom sprint backlogen blir tømt, vil Scrum Master i samråd med teamet hente flere elementer fra product backlogen. Dersom vi ikke klarer å bli ferdige med sprint backlogen vil uferdige element bli returnert til product backlogen og revurdert ved neste sprint planning meeting.

Plan for statusmøter og beslutningspunkter

Daglige Scrum møter vil holdes hver dag kl 10:00 utenom mandag der de vil bli holdt kl 11:00 på grunn av forelesninger. Disse møtene vil normalt vare 10-15 minutter der alle informerer resten av gruppen om hva de fikk gjort, eventuelle problemer og hva de har planlagt å gjøre i løpet av dagen.

Det er planlagt med Nammo å holde et møte på deres lokaler to ganger i måneden, det er ikke satt av en bestemt dag eller dato. Møtene blir satt opp når vi er ferdig med en sprint og det kan planlegges ekstra møter hvis det oppstår uklarheter underveis som ikke kan løses gjennom e-post.

Det er satt opp møte med veileder hver uke på tirsdag kl 13:00 der vi vil gå gjennom hva som er blitt gjort og hva som er planlagt. Vi vil også ta opp eventuelle tekniske eller programmatiskke problemer vi har problemer med her.

Organisering av kvalitetssikring

Dokumentasjon, Standardbruk og kildekode

Verktøyene vi bruker for å holde kildekoden godt organisert og dokumentert er:

Code review

All kildekode på ferdig programmert funksjonalitet går gjennom sammen for å forsikre en god kodestandard og minimere sjansen for feil eller bugs.

Forhåndsbestemt formatering

Vi vil følge formateringen som er anbefalt fra python.org; PEP 8³. Denne dekker de fleste eventualiteter veldig godt og sørger for god leselighet og konsistens i all kode. Dersom det oppstår noen uklarheter om det som står i veiledningen eller noe ikke er dekket godt nok der, vil det bli diskutert i gruppen og vi vil i fellesskap bestemme hvilken standard vi vil bruke.

Dokumentasjon

All kode skal kommenteres og i toppen av hver fil vil det være en kort beskrivelse av hva den gjør og eventuelle anbefalinger i relasjon til bruk av filen.

Verktøy

ShareLaTeX

Vi har valgt å dokumentere alt i shareLaTeX ettersom det er et verktøy som ga oss muligheten til å jobbe på samme dokumentet samtidig.

Git og Bitbucket

Vi har valgt å bruke Git som vår versjonskontroll, og bruker bitbucket for å host respoetory. Reposetory skal være lukket gjennom utviklingsprosessen, om det blir gjort tilgjengelig etter oppgaven er ferdig er Nammo sitt valg. Når det kom til valget av versjonskontroll var det en rask avslutting. Bitbucket har blitt brukt mye gjennom vår utdanning og er den versjonskontrollen vi har mest kjennskap til. Bitbucket er blitt en industristandard som vi og vår arbeidsgiver mente var et godt valg.

Eclipse med PyDev

Utviklingsmiljøet vil være Eclipse med utvidelsen PyDev installert. Eclipse gir gode muligheter for egendefinering av innstillinger, syntax-belysning og gruppen har tidligere erfaring med bruk av verktøyet.

Toggle

Toggle er et verktøy for å registrere tidsbruk på prosjekter, og gir gode diagrammer for analyse av hva som krever mye eller lite tid. Toggle kom sterkt anbefalt av kolleger og vi har hatt gode erfaringer med verktøyet selv.

Trello

Trello vil bli brukt for organisering av product backlog og gir gode muligheter for å definere arbeidsflyt og holde styr på hva som er hvor i utviklingsprosessen. Vi har jobbet med verktøyet før og er vel fornøyd med det.

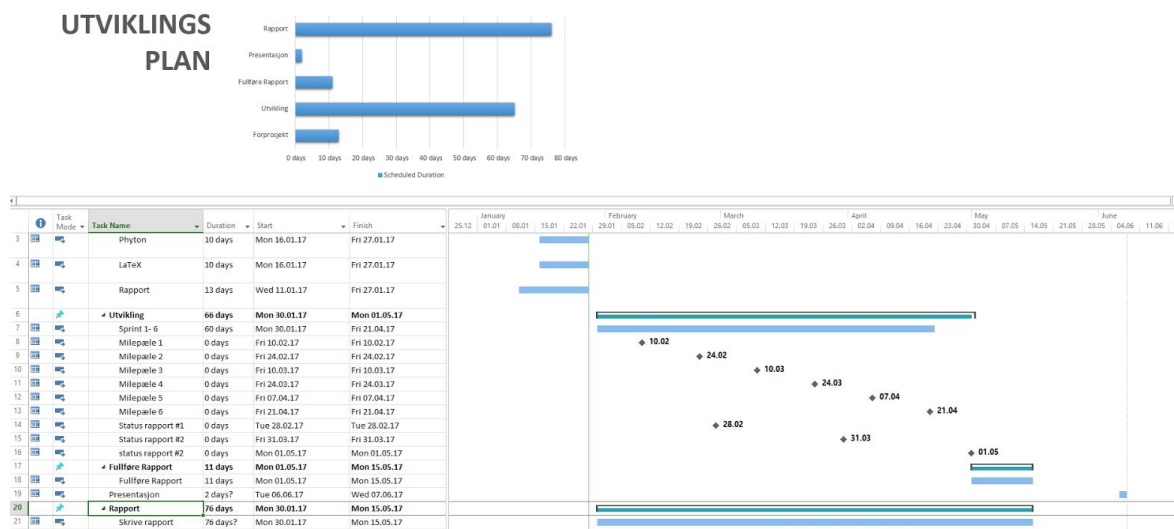
Risikoanalyse

Fargekode			
Usannsynlig	Lite sannsynlig	Sannsynlig	Svært sannsynlig
Ufarlig	Truende	Risikabel	Katastrofal

NR	Beskrivelse	Sannsynlighet	Risiko	Tiltak
1.	Tap av maskinvare	Sannsynlig	Risikabel	Oppgaven skrives i ShareLatex og sendes til Github.
2.	Langtidssykdom	Lite sannsynlig	Katastrofal	Jobbe tett sammen så resterende utvikler kan ta over arbeidet til sykt medlem
3.	Problemer med implementasjon av andre systemer	Sannsynlig	Ufarlig	
4.	Forsinkelser	Svært sannsynlig	Truende	Planlegge rom for forsinkelser
5.	Mangelfull funksjonalitet	Lite sannsynlig	Risikabel	
6.	Tap av informasjon	Lite sannsynlig	Ufarlig	Oppgaven vil ikke omhandle noe sensitiv informasjon

Plan for gjennomføring

Gantt diagram



Utviklingsfasen er delt inn i seks sprinter der milepælene markerer slutten av hver sprint, men slutten på hver sprint vil kunne variere noe etter når Nammo har tid til å møte med oss.

Ressursplan

Denne seksjonen gir et overblikk over de forskjellige arbeidsoppgavene som vil bli fokusert på under arbeidet. Denne oppdeling er en approksimasjon og skal brukes som en retningslinje, det er sannsynlig at det vil bli endringer etter hvert som vi jobber.

Forprosjektet

Under dette stadiet av prosjektet var det ikke blitt delt inn noen forskjellige oppgaver på gruppemedlemmene. Medlemmene skulle sammen fokusere på å sette seg inn i oppgaven og hvilke verktøy som kunne bli brukt under dette prosjektet. Det vil da bli jobbet ca 70% av tiden på forprosjektet og arbeid med planlegging. De resterende

30% av tiden vil bli brukt til å sette seg inn i verktøy som Latex og Eclipse, og å lære mer om python og hvilke bibliotek som vil bli brukt under utvikling.

Utviklingsfasen

Når gruppen har kommet i gang med utviklingen skal tiden fordeles med 20% av tiden for rapportskriving og 70% for utvikling av programvaren, de resterende 10% forventes å forsvinne i møter og andre diskusjoner innad i gruppen. For at rapporten skal bli så omfattende og detaljert som mulig skal alle beslutninger noteres med begrunnelse og vurderte alternativer. Disse notatene skal utvides på når det skrives rapport, så minst mulig blir glemt.

Leveringsfase

Leveringsfasen begynner 11 dager før innlevering, i denne fasen skal all kode være ferdig og det eneste som må gjøres er å rette på eventuelle bugs. Vi planlegger at det blir lite arbeid med kildekode i denne fasen og forventer at minst 70% av tiden vil gå til rapportskriving, med resterende tid fordelt mellom bugfiksing og kommunikasjon med arbeidsgiver og veileder.

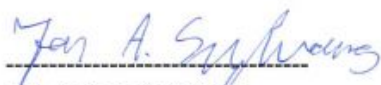
Referanser

1. Nammo AS. About us [website]. Nammo AS; 2017 [updated 2017; cited 2017.01.23]. Available from: <https://www.nammo.com/who-we-are/about-us/>
2. Stephen R. Palmer. Feature-Driven Development (FDD) [website]. step-10.com; 2017 [updated 2017; cited 2017.01.17]. Available from: <http://www.step-10.com/SoftwareProcess/FeatureDrivenDevelopment/index.html>
3. Guido van Rossum, Barry Warsaw, Nick Coghlan. PEP 8 -- Style Guide for Python Code [website]. python.org; 2017 [updated 2017; cited 2017.01.25]. Available from: <https://www.python.org/dev/peps/pep-0008/>

A Grupperegler

Gruppe regler for Bacheloroppgave

- Alle medlemmer får 3 veto rettigheter som kan brukes når som helst under prosessen. Disse kan bli brukt når gruppen kommer til en uenighet. Det kan kunne brukes en veto rett per problem. Hvis alle medlemmer i gruppe skulle bruke opp alle veto rettighetene vil de får utdelt 3 nye. Veto rettingehetene kan ikke brukes ved utkastings- og kostnadsavgjørelser.
- Det er ingen behov for en prosjektleder ettersom størrelsen på gruppe er så liten.
- Martin Holltrø Spongsveen står som kontaktperson og har rettigheten til å signere for gruppen.
- Hvis en av gruppemedlemmene ikke stiller med ferdig arbeid til tidsfrist vil han bli varslet, hvis medlemmet ikke blir ferdig med arbeidet etter å ha bli varslet kan det føre til potensiell utkasting i samråd med veileder.
- Grunner for utkasting er inaktivitet over en lengre periode.
- Utkasting skjer i samråd med veileder.
- Dersom det påløper kostnader under utviklingsprosessen skal de splittes likt mellom medlemmene.


Jon Anders Sylvarnes


Martin Holltrø Spongsveen
