



**FAKULTA
STROJNÍ
ČVUT V PRAZE**

Ústav automobilů, spalovacích motorů
a kolejových vozidel

Vývoj globálně-optimálního řídicího algoritmu
hybridního vozidla

Development of a globally optimal powertrain
control algorithm

DIPLOMOVÁ PRÁCE
2022

Bc. Jan Filip

Studijní program: NDTT DOPRAVNÍ A TRANSPORTNÍ TECHNIKA
Studijní obor: N071TMVO Motorová vozidla
Vedoucí práce: Ing. Rastislav Toman

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Filip** Jméno: **Jan** Osobní číslo: **473466**
Fakulta/ústav: **Fakulta strojní**
Zadávající katedra/ústav: **Ústav automobilů, spalovacích motorů a kolejových vozidel**
Studijní program: **Dopravní a transportní technika**
Specializace: **Motorová vozidla**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Vývoj globálně-optimálního řídicího algoritmu hybridního vozidla

Název diplomové práce anglicky:

Development of a globally optimal powertrain control algorithm

Pokyny pro vypracování:

1. Zpracujte rešerši zaměřenou na optimální metody řízení hybridního vozidla.
2. Navrhněte globálně-optimální algoritmus používající princip dynamického programování (DP) a simulujte jeho funkci - s použitím programovacího jazyka Python.
3. Navrhněte a testujte možnosti zlepšení výpočetního času základního algoritmu DP, při zachování (zlepšení) jeho přesnosti.
4. Připravte porovnání vyvinutého DP algoritmu s implementací DP v SW GT-Suite.

Seznam doporučené literatury:

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Rastislav Toman ústav automobilů, spalovacích motorů a kolejových vozidel FS

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **20.04.2022**

Termín odevzdání diplomové práce: **13.07.2022**

Platnost zadání diplomové práce: _____

Ing. Rastislav Toman
podpis vedoucí(ho) práce

doc. Ing. Oldřich Vitek, Ph.D.
podpis vedoucí(ho) ústavu/katedry

doc. Ing. Miroslav Španiel, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta



Anotační záznam

Jméno autora: Jan Filip

Název práce: Vývoj globálně-optimálního řídicího algoritmu hybridního vozidla

Anglický název: Development of a globally optimal powertrain control algorithm

Rozsah práce:

- 81 stran
- 20 obrázků
- 10 tabulek
- 24 grafů

Akademický rok: 2021/2022

Ústav: 12 120 Ústav automobilů, spalovacích motorů a kolejových vozidel

Studijní program: Dopravní a transportní technika

Vedoucí práce: Ing. Rastislav Toman

Klíčová slova: Hybridní vozidlo, Dynamické programování, Globálně-optimální algoritmus, Strategie řízení

Key words: Hybrid electric vehicle, Dynamic programming, Globally optimal algorithm, Energy management strategy



Abstrakt

Hybridní pohonný systém je jednou z cest ke snížení spotřeby paliva a emisí oxidu uhličitého u konvenčních vozidel. Aby byl využit co nejlépe potenciál hybridního pohonu, je nutné implementovat strategii řízení, která rozděluje požadovaný točivý moment mezi spalovací motor a elektromotor. Tato práce se zabývá vývojem a optimalizací strategie řízení hybridního pohonu s využitím dynamického programování. Strategie byla otestována na několika jízdách a v závěru byl řídicí algoritmus porovnán s implementací dynamického programování v programu GT-Suite.

Abstract

A hybrid propulsion system is one of the options to reduce fuel consumption and vehicle emissions for conventional vehicles. To exploit the potential of a hybrid powertrain to the biggest extent, it is necessary to implement a control strategy to distribute required torque between the internal combustion engine and the electric motor. Therefore this thesis deals with development and optimization of a hybrid powertrain control strategy based on dynamic programming. The strategy was tested on several driving cycles and in the end the control algorithm was compared with the implementation of dynamic programming in GT-Suite.



Čestné prohlášení

Prohlašuji, že jsem diplomovou práci na téma: “ Vývoj globálně-optimálního řídicího algoritmu hybridního vozidla ” vypracoval samostatně s použitím odborné literatury a pramenů, uvedených v seznamu zdrojů, jenž tvoří poslední kapitolu této práce.

V Praze dne:

.....

Jan Filip



Poděkování

Tímto bych chtěl poděkovat svému vedoucímu, Ing. Rastislavu Tomanovi, za jeho vedení, podporu a užitečné rady při tvorbě této diplomové práce.



Obsah

Úvod.....	9
1 Hybridní vozidla	11
1.1 Topologie hybridního pohonu	12
1.2 Stupeň elektrifikace	16
2 Strategie řízení hybridního pohonu.....	19
2.1 Heuristické metody.....	21
2.2 Optimalizační metody.....	24
2.2.1 Dynamické programování.....	26
2.2.2 Pontrjaginův princip minima	28
2.3 Prediktivní a adaptivní strategie řízení.....	29
3 Vývoj řídicího algoritmu.....	31
3.1 Model vozidla	32
3.1.1 Parametry vozidla a jeho konfigurace.....	33
3.1.2 Výpočet požadované trakční síly.....	34
3.1.3 Výpočet požadovaného točivého momentu	37
3.1.4 Baterie	38
3.1.5 Elektromotor.....	40
3.1.6 Spalovací motor	43
3.2 Strategie řazení.....	44
3.3 Jízdní cykly	46
3.4 Algoritmus dynamického programování	49
3.4.1 Odvození algoritmu	49
3.4.2 Výpočetní proces	52
3.4.3 Implementace algoritmu na řízení hybridního pohonu	56
4 Metody zlepšení výpočetního času základního algoritmu	60
4.1 Nastavení limit pro výpočet pouze s přípustnými hodnotami	60
4.2 Memorizace	62



4.3 Nahrazení map účinností pomocí aproximace polynomem.....	63
5 Výsledky simulace.....	65
6 Porovnání vyvinutého algoritmu s implementací dynamického programování v programu GT-Suite.....	71
6.1 Princip výpočetního procesu	71
6.2 Simulace.....	72
Závěr	75
Seznam použitých informačních zdrojů.....	76
Seznam obrázků.....	79
Seznam grafů	80
Seznam tabulek	81



Úvod

Vzhledem k tomu, že spotřeba neobnovitelných zdrojů energie v podobě fosilních paliv a zvyšující se emise oxidu uhličitého, jednoho z plynů přispívajících k tzv. skleníkovému efektu, začínají být v současné době stále větším problémem pro lidskou společnost, probíhají v posledních letech významné změny na poli dopravy a transportu lidí. Emise oxidu uhličitého jsou úzce spojeny se spotřebou paliva, a proto je v současnosti cílem vývoje v oblasti automobilového průmyslu vytvoření pohonných systémů s co největší účinností, s použitím nových technologií, nebo více druhů pohonných jednotek, které spolu navzájem spolupracují. K těmto účelům jsou ve velké míře používány elektrické stroje. Trendem převážně posledních let jsou elektromobily, ve kterých je spalovací motor kompletně nahrazen elektromotorem. Z důvodu velkých pořizovacích nákladů, nízkého dojezdu v porovnání s konvenčním vozidlem a neuspokojivou infrastrukturou nabíjecích stanic je ale prozatím zastoupení čistě elektrických vozidel pouze v rámci jednotek procent. Výhodnou alternativou mohou být hybridní vozidla.

Hybridní vozidla, která v sobě spojují tradiční spalovací motor a elektrický pohonný systém, představují dobrý kompromis mezi konvenčním vozidlem a elektromobilem, za předpokladu spolupráce obou pohonných systémů. Díky tomu dojde ke zlepšení účinnosti spalovacího motoru, zatímco není nutnost velkých a drahých baterií, jako je tomu u elektrických vozidel. Zařazení elektromotoru do pohonného systému umožní používat spalovací motor převážně v pracovních bodech s vyšší účinností, zatímco v bodech s nízkou účinností je zastoupen elektromotorem, který není tak závislý na provozním rozsahu, a to ve velké míře napomáhá ke snižování spotřeby paliva. Jednou z hlavních výhod je možnost rekuperace kinetické energie, když vozidlo brzdí, což v konečném důsledku také přispívá ke snížení spotřeby paliva.

Zařazením elektromotoru do pohonného systému se ale na druhou stranu stává pohonný systém mnohem komplexnější a složitější na řízení. Jelikož jsou zde přítomny dva zdroje energie, je nutné řídit, kdy se bude jaký používat. To samozřejmě není možné řídit manuálně řidičem, ale musí k tomu být navržen automatický řídicí systém a řídicí strategie, aby bylo dosaženo ideálního rozdělení toku točivého momentu mezi spalovacím motorem a elektromotorem, což vede k požadovanému snížení spotřeby energie. Existuje několik různých strategií řízení, od jednodušších, založených například na bázi pravidel, které berou



v potaz pouze aktuální stav vozidla, až po složitější strategie, jako například optimalizační metody, které rozhodují jak na základě aktuálního stavu vozidla, tak zohledňují i minulá data a predikci budoucího stavu v daném jízdním cyklu.

Tato práce se bude zabývat optimalizačními metodami řízení pohonného systému hybridních vozidel. Půjde převážně o vývoj a optimalizaci globálně-optimálního řídicího algoritmu hybridního vozidla, ve kterém bude využíváno dynamického programování.

Struktura práce bude vypadat následujícím způsobem – v prvních kapitolách bude nastíněn obecný základ o tématu hybridních vozidel, následovaný detailnějším přiblížením jednotlivých strategií řízení pohonného systému u hybridních vozidel. V dalších kapitolách už bude podrobně znázorněn vývoj algoritmu řízení a vše s tím spojené. V posledních kapitolách budou uvedeny výsledky simulací, které byly naprogramovány pomocí programovacího jazyka Python, a vytvořený algoritmus bude porovnán s implementací dynamického programování v programu GT-Suite.



1 Hybridní vozidla

Hybridní vozidlo je podle definice takové vozidlo, které je vybaveno nejméně dvěma zásobníky energie a jim příslušející hnací agregáty, které přeměňují tuto energii na trakční práci. Bývá zvykem, že jeden z těchto zásobníků je možné dobíjet během jízdy. Ve většině případů se jedná o kombinaci pohonného systému se spalovacím motorem doplněným o alternativní zdroj energie. Tím může být například elektrická energie uložená v bateriích, případně superkapacitorech, potenciální energie, nebo kinetická energie. [1] [2]

Tato práce se bude zabývat vývojem řídicího algoritmu pro hybridní elektrické vozidlo, tedy kombinaci pohonu se spalovacím motorem a elektromotorem, což je mimo jiné i nejčastější varianta hybridního pohonu. Proto bude v následujících kapitolách probírána pouze tato varianta.

Jednou z hlavních motivací pro vytvoření vozidla s hybridním pohonem je spojení výhod konvenčních vozidel se spalovacím motorem, jako například udržení vysokých otáček, nebo vysoká energetická hustota paliva, s výhodami elektromobilů, jako třeba prakticky nulové lokální emise, nebo možnost maximálního točivého momentu už od nulových otáček motoru. Další zásadní motivací je snížení spotřeby paliva v porovnání s vozidlem se spalovacím motorem. V závislosti na typu hybridního vozidla je možné dosáhnout i následujících bodů:

- Rekuperace kinetické energie při brzdění vozidla do baterie, namísto disipace při tření v brzdách
- Pomocí řídicí jednotky optimalizovat rozdělení výkonu mezi jednotlivými pohonnými systémy
- Downsizing spalovacího motoru při zachování dostatečného výkonu pro splnění jízdních požadavků
- Systém start-stop, tedy možnost vypnutí spalovacího motoru, když není potřeba výkon a tím zamezit spotřebě paliva při volnoběžných otáčkách.

Na druhou stranu je ale nutné počítat s tím, že z důvodu složitějšího pohonného systému a vybavením vozidla baterií, je hmotnost hybridního vozidla o 10–30% větší než u konvenčního vozidla. [1]



Hybridní vozidla je možné rozdělit podle různých kritérií, kde ale většinou nejsou stanovené žádné přesné hranice. Nejčastěji se hybridní vozidla rozdělují na základě následujících kritérií:

- Podle konstrukčního uspořádání pohonného systému (topologie)
- Podle stupně elektrifikace – vyjadřuje jaké množství energie je dodáváno elektrickou částí pohonného systému

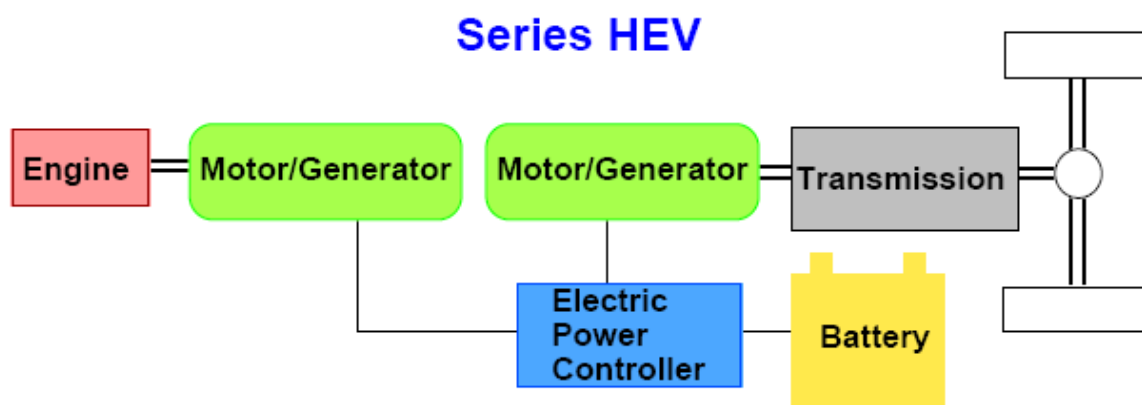
1.1 Topologie hybridního pohonu

Prvním kritériem pro rozdělení hybridních vozidel je uspořádání jejich pohonného systému. Jeho součástí většinou bývá klasický spalovací motor a alespoň jeden elektrický stroj. Na základě uspořádání těchto základních prvků v pohonném systému vzniká velké množství variant, které lze rozdělit do následujících skupin:

- Sériový hybrid
- Paralelní hybrid
- Kombinovaný (série-paralelní) hybrid

Sériový hybrid

Schématické znázornění sériového hybridního pohonu je zobrazeno na obr. č. 1.



Obrázek 1: Sériový hybridní pohon [14]



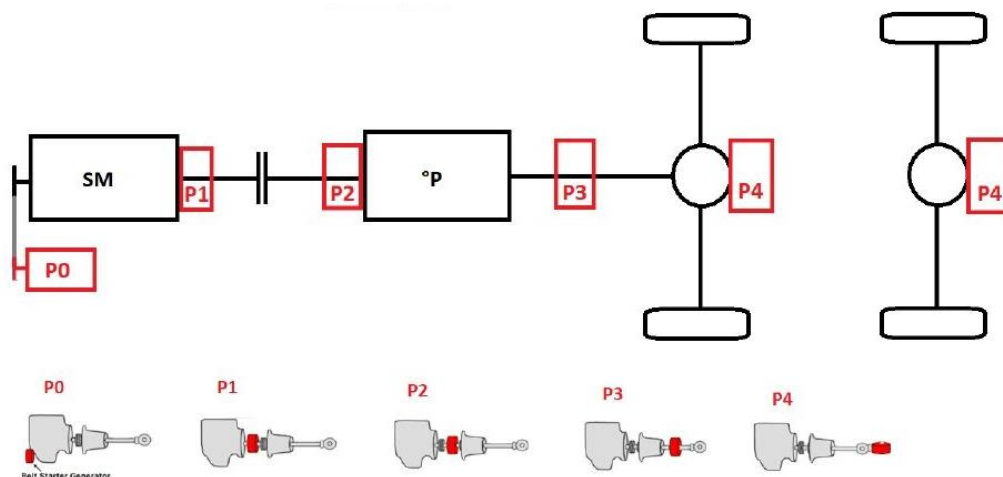
Sériové hnací ústrojí je nejjednodušší konfigurací hybridního pohonu. Spalovací motor nemá žádné mechanické spojení s poháněnými koly a to znamená, že veškerou trakční sílu dodává elektromotor. Spalovací motor není závislý na výkonnostních požadavcích vozidla a otáčkách kol, a proto je možné ho udržovat v optimálních pracovních bodech, což přispívá k nižší spotřebě paliva a emisím. Používá se výhradně pro pohánění generátoru, který pak přímo napájí trakční elektromotor, v případě velkého zatížení, nebo nabíjí baterii. Baterii je možné dobíjet i pomocí elektromotoru při brzdění rekuperací. Při zvolení elektromotoru s vhodnou charakteristikou navíc není potřeba převodovky, protože elektromotor dokáže pracovat v širokém pásmu otáček. Je možné také použít více elektromotorů a tím pohánět každé kolo samostatně. Jednou z nevýhod tohoto uspořádání je dvojnásobná změna energie mezi spalovacím motorem a poháněnými koly. Nejdříve mezi spalovacím motorem a generátorem a následně mezi generátorem a elektromotorem. Díky tomu dochází ke ztrátě energie a snížení účinnosti. [4] [5] [29]

Sériová hybridní vozidla jsou nejučinnější na jízdních cyklech, kde je nutné časté zastavování a rozjíždění. Proto jsou ideální do městského provozu a jsou využívána například jako doručovací vozidla, nebo městské autobusy. Příkladem hybridního vozidla se sériovým pohonem je například Nissan Qashqai e-Power. [3]

Paralelní hybrid

Další možností je paralelní hybridní pohon. Už z názvu je zřejmé, že v této konfiguraci bude elektromotor, nebo více elektromotorů, dodávat hnací moment paralelně se spalovacím motorem. Elektromotor je se spalovacím motorem spojen mechanickou vazbou a oba jsou závislé na aktuálních otáčkách kol. U většiny paralelních hybridních vozidel se elektromotor a generátor slučují do jedné jednotky. Oproti sériovým hybridům je v tomto uspořádání pouze jedna přeměna energie, takže nebude docházet k takovým ztrátám a bude zde vyšší účinnost pohonného ústrojí. Příkladem hybridního vozidla s paralelním pohonem je například Honda Insight nebo Civic. [3] [29]

Na základě umístění elektromotoru v pohonném systému je možné rozdělit paralelní hybridy na další podskupiny. Tyto podskupiny jsou ve většině případů označovány jako P_n , kde n je číslo od 0 do 4, které vyjadřuje umístění elektromotoru. Jednotlivé možnosti jsou schematicky znázorněny na obr. č. 2.



Obrázek 2: Umístění elektromotoru v pohonném systému paralelního hybridu [29]

P0 paralelní hybrid je cenově efektivní řešení pro hybridizaci stávajících pohonných systémů poháněných pouze spalovacím motorem. Elektromotor-generátor je se spalovacím motorem spojený pomocí hnacího řemene a zastává funkci alternátoru a startéru, stejně tak jako hybridní funkce asistence spalovacímu motoru. Toto řešení je velmi zastoupené především u mild hybridů.

P1 paralelní hybrid má elektromotor-generátor na místě setrvačnicku a je přímo spojený s hnací hřídelí. Oproti variantě P0 je výhodou odstranění spojení pouze přes hnací řemen. Toto řešení je opět určeno spíše pro mild hybridy, protože výkon elektromotoru je limitován zástavbovými možnostmi. Nicméně P1 hybridy nejsou moc používány, především z důvodu nutnosti výrazných změn v zástavbě a pouze malého zlepšení co se týče spotřeby paliva.

P2 paralelní hybrid má elektromotor-generátor umístěný mezi spalovacím motorem a převodovkou, nebo může být i její součástí. Pomocí spojky je možné odpojit spalovací motor a tím dosáhnout čistě elektrického pohonu, narozdíl od předchozích dvou variant. Jelikož je část výkonu dodávána elektromotorem, je možné udržovat spalovací motor v pracovních bodech s vyšší účinností. K tomu je nutný řídicí algoritmus. V případě brzdění rekuperací pracuje elektromotor jako generátor a nabíjí baterii. P2 hybridy jsou v současnosti nejvíce zastoupenou variantou z paralelních hybridů.

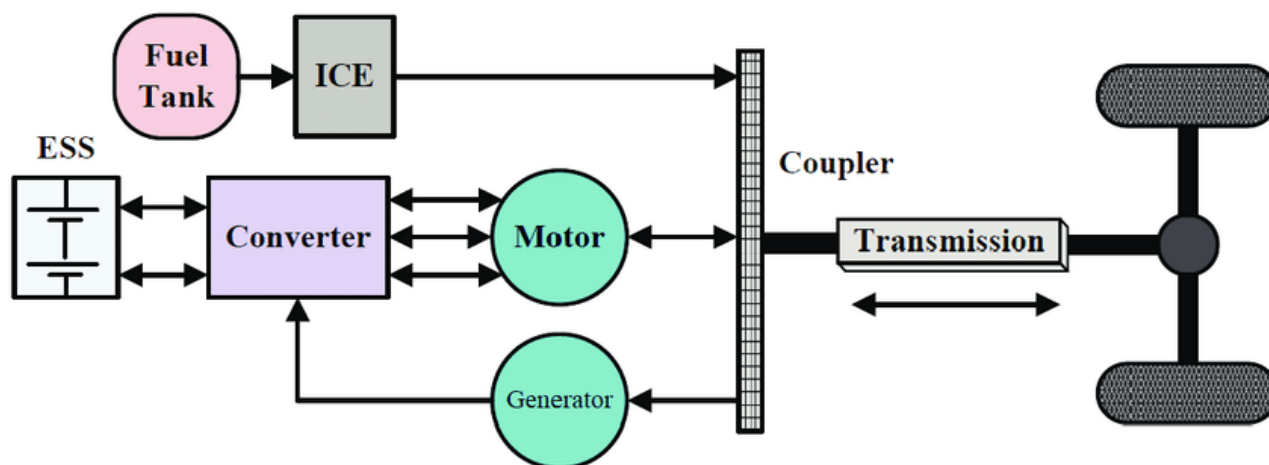
P3 paralelní hybrid má elektromotor-generátor připojen přímo na hnací hřídel mezi převodovkou a diferenciálem, nebo nápravou. To samozřejmě znamená to, že elektromotor musí být schopný pokrýt velké spektrum otáček a momentu. Z toho důvodu je potřeba používat větší a výkonnější elektromotory. Celkově je ale uspořádání P3 velmi málo zastoupené.



P4 paralelní hybrid má elektromotor-generátor umístěný u nápravy nebo přímo v kolech. S tím je opět spojeno to, že elektromotor musí být schopný pokrýt velké spektrum otáček a vysoké momenty. Na druhou stranu je ale účinnost elektrického pohonu nejvyšší ze všech variant paralelních hybridů. Existuje také varianta, kde jsou elektromotory umístěny na druhé nápravě, která je označována TTR (Through the road).

Kombinovaný hybrid

Kombinovaný hybrid může být označován také jako sério-paralelní, nebo power-split hybrid. Konfigurace pohonného systému je schematicky znázorněna na obrázku č. 3. Sériové a paralelní uspořádání hybridního pohonu mají každá své výhody a preferované použití, jako například pro přerušovanou jízdu s nízkou rychlostí ve městě je výhodnější sériový hybrid, zatímco při jízdě vyšší rychlostí mimo město je zase výhodnější hybrid paralelní. Z toho důvodu je snaha tyto výhody zkombinovat pro jedno uspořádání hybridního pohonu, kterým je právě kombinovaný, sério-paralelní, hybrid. Jedná se převážně o paralelní uspořádání s prvky sériové konfigurace.



Obrázek 3: Kombinovaný hybrid [15]

Kombinovaný hybridní pohon využívá jak mechanickou, tak elektrickou vazbu mezi jednotlivými prvky pohonného systému. Zároveň se vyznačuje použitím dvou elektromotor-generátorů. Jeden je, podobně jako u paralelního uspořádání, využíván jako trakční motor, nebo pro rekuperaci při brzdění, zatímco ten druhý je využíván, stejně jako u sériové konfigurace, jako generátor pro nabíjení baterie pomocí spalovacího motoru, nebo k zajištění funkce start-stop. Ve většině případů je spojení spalovacího motoru a dvou elektromotorů dosaženo pomocí planetového děliče, který rozděluje hnací moment na základě aktuálních požadavků. Jinou variantou je možnost přepínání mezi sériovým a paralelním zapojením, přidáním dvojice spojek. [1] [29]



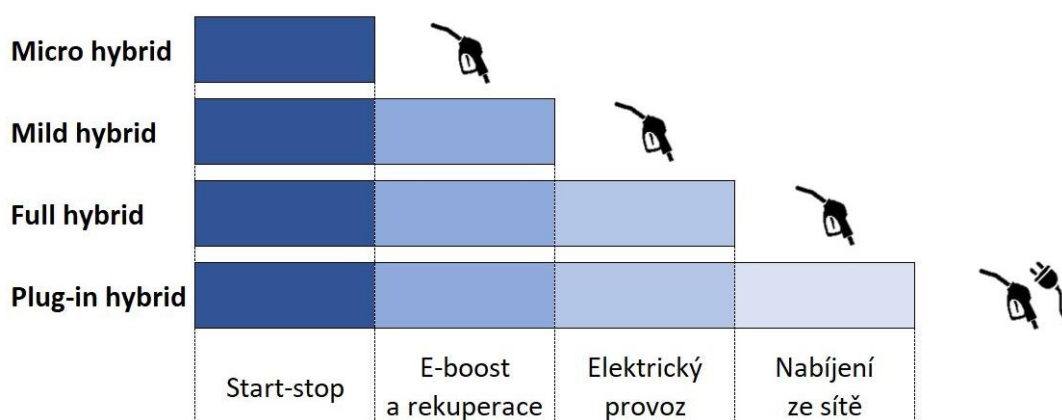
Kombinovaný hybridní pohon je samozřejmě mnohem komplikovanější než uspořádání sériové, či paralelní, a s tím také přichází vyšší cena tohoto pohonného systému. Zároveň je kvůli větší komplexnosti potřeba výkonnější řídicí systém. Takovéto uspořádání hybridního pohonu se používá například ve vozech Toyota Prius. [2]

1.2 Stupeň elektrifikace

Druhé kritérium pro rozdělení hybridních vozidel je podle stupně elektrifikace. Jak už bylo zmíněno, hybridní vozidla zpravidla využívají pohonný systém se spalovacím motorem a alespoň jedním elektromotorem. Na základě podílu využití elektrické energie v pohonném systému je možné hybridní vozidla rozdělit do následujících skupin:

- Micro hybrid
- Mild hybrid
- Full hybrid
- Plug-in hybrid

Skupiny jsou seřazeny podle využití elektrické energie v pohonném systému od nejmenšího zastoupení po největší. Co se týče nabíjení baterie, tak první tři skupiny jsou ještě plně závislé na spalovacím motoru, nebo rekuperaci při brzdění, zatímco jediné u plug-in hybridu je možné baterii dobíjet přímo z elektrické sítě. Jednoduché porovnání a ukázka základních vlastností jednotlivých skupin je zobrazeno na obrázku č. 4. Důkladnější rozbor každé varianty bude popsán v následující sekci.



Obrázek 4: Stupeň elektrifikace hybridních vozidel



Micro hybrid

Micro hybridy jsou nejméně elektrifikovaná skupina hybridních vozidel. Jde v podstatě o konvenční vozidla se spalovacím motorem, která jsou vybavena posíleným elektrickým příslušenstvím, jako například startérem, alternátorem a baterií, a pokročilým systémem start-stop. Systém start-stop umožňuje na základě požadavků z řídicí jednotky automatické vypínání spalovacího motoru při dojezdu a při zastavení a následné velmi rychlé nastartování motoru před rozjezdem, čímž se šetří palivo, které by bylo ztraceno při chodu motoru bez zátěže. Z tohoto důvodu disponují micro hybridy předimenzovaným startérem o výkonu 3 až 15 kW a napětí 12 V, který zajišťuje rychlé a hladké nastartování motoru. I přes označení hybrid, nejsou tato vozidla schopna využívat elektrickou energii ke svému pohybu. Dokáží pouze v omezené míře rekuperovat brzdovou energii prostřednictvím alternátoru. [3] [6] [29]

Micro hybridy jsou nejvíce zastoupeny jako malá osobní vozidla určená především do městského provozu, kde se nejlépe využije jejich vlastností. Příkladem micro hybridu může být třeba Mercedes Smart.

Mild hybrid

Další skupinou na stupnici elektrifikace jsou mild hybridy. Ty opět, stejně jako micro hybridy, nejsou schopné čistě elektrického provozu. Elektromotory v těchto vozidlech nemají dostatečný výkon pro zajištění jízdy čistě na elektrickou energii, kromě rozjezdu a jízdy nízkou rychlostí – do 20 km/hod. Jejich funkcí je asistence spalovacímu motoru při zrychlování a vysokých výkonech, rekuperace energie při brzdění a již zmíněný rozjezd vozidla. Mild hybridy mají většinou 48 voltový systém namísto 12-ti voltového, jako tomu bylo například u micro hybridů. Jelikož je v těchto vozech elektromotor převážně jenom jako asistent pro spalovací motor, vystačí si mild hybridy pouze s malou baterií, běžně s kapacitou okolo 0,5 kWh. Ta je dobíjena buď pomocí spalovacího motoru, nebo rekuperací při brzdění. [3] [7] [29]

Mild hybridy jsou v současnosti u výrobců velmi oblíbeným typem hybridního pohonu. Výhodou je menší konstrukční náročnost, poměrně nízká cena a hmotnost. I přes použití elektromotorů s nízkým výkonem a baterií s malou kapacitou, dosahují mild hybridy relativně dobré úspory ve spotřebě paliva. Hybridní pohony ve formě mild hybridu mají například vozy Volkswagen Golf, nebo Audi A4.



Full hybrid

Full hybridy jsou první skupinou, která je schopna elektrického provozu, na rozdíl od micro a mild hybridů. Je možné jet pouze s pohonem spalovacího motoru, nebo jen s elektromotorem, nebo kombinací obou, kde elektromotor asistuje spalovacímu motoru. Elektromotor je využíván opět k rozjezdu, k asistenci spalovacímu motoru při vysokých výkonech, nebo k jízdě v místech, kde je zakázané používání spalovacích motorů. Jelikož jsou ale u full hybridů používány stále ještě relativně malé baterie, obvykle s kapacitou okolo 1,5 kWh, je možné využívat čistě elektrický provoz pouze v rámci jednotek kilometrů. Nabíjení baterie probíhá opět pomocí rekuperace při brzdění, nebo pomocí spalovacího motoru. [29]

Co se týče snižování spotřeby, tak full hybridy jsou opět o něco výhodnější než mild hybridní vozidla, kde největší úspora bude znovu v městském provozu. Příkladem full hybridů je například Toyota Prius a Yaris.

Plug-in hybrid

Poslední skupinou jsou plug-in hybridní vozidla. Českým ekvivalentem je hybridní automobil s dobíjením ze sítě, takže už z názvu je zřejmý hlavní rozdíl oproti předchozím skupinám. Tím je tedy možnost dobíjení baterie z externího zdroje energie, typicky z elektrické sítě. I přes to je ale stále využíváno dobíjení baterie rekuperací při brzdění a pomocí spalovacího motoru, jako tomu bylo například u mild a full hybridů. [3] [8] [29]

Co se týče elektrického pohonu, tak jsou plug-in hybridy v podstatě velmi podobné full hybridním automobilům, tedy že jsou také schopny čistě elektrického pohonu, s tím rozdílem, že je možné je dobíjet z elektrické sítě. V porovnání s full hybridy je většinou používán menší spalovací motor, a naopak větší a výkonnější elektromotor a několikanásobně větší baterie, s kapacitou typicky 10 – 15 kWh. Díky tomu se prodloužil dojezd na čistě elektrický pohon na desítky kilometrů, běžně do 50 kilometrů. Díky většímu elektromotoru mají plug-in hybridy také vyšší schopnost rekuperativního brzdění než ostatní hybridy. [3] [8] [29]

Jelikož dokáží plug-in hybridy ujet až desítky kilometrů na čistě elektrický pohon, jsou ideálním prostředkem pro každodenní dojíždění do práce, nebo jízdu po městě, ale zároveň i na občasné dlouhé cesty za použití klasického hybridního módu. Hlavní nevýhodou těchto vozidel je vysoká cena, a to převážně z důvodu použití větších baterií a elektromotoru a složitosti pohonného systému. Mezi zástupce plug-in hybridů patří například Škoda Octavia iV a Superb iV, nebo Hyundai Ioniq.



2 Strategie řízení hybridního pohonu

Automatický řídicí systém a strategie řízení hybridního pohonu je nezbytnou součástí každého hybridního vozidla pro zajištění jeho správného a požadovaného chování. Strategie řízení hybridního pohonu je jednou ze tří základních fází při hledání nejefektivnějšího způsobu, jak se dostat hybridním vozidlem z počáteční do koncové polohy zvolené cesty. První fáze se označuje anglickým názvem eco-routing a představuje výběr nejvhodnější cesty z hlediska spotřeby energie. Druhá fáze se nazývá eco-driving, nebo někdy bývá označována českým názvem optimalizace průběhu jízdy. V této fázi jde o nalezení energeticky nejefektivnější jízdní strategie a rychlostního profilu. Poslední fází je již zmíněná strategie řízení hybridního pohonu, která v podstatě rozděluje výkon mezi spalovací motor a elektromotor na základě požadovaného výkonu. Ten je v každém okamžiku jízdy určen aktuální rychlostí vozidla a jízdními odpory.

Mezi základní cíle a úlohy strategie řízení hybridního pohonu patří například následující činnosti:

- Nalezení vhodného způsobu řízení pro dosažení minimální spotřeby a emisí při zachování potřebných vlastností vozidla
- Optimální rozdělení požadovaného výkonu mezi spalovací motor a elektromotor
- Udržení stavu nabití baterie uvnitř stanovených mezí
- Nebo případně výběr optimálního převodového stupně.

Tato práce se bude zabývat výhradně jen tou poslední fází, tedy strategiemi řízení. Co se týče rozdílných topologií hybridních vozidel, zmíněných v předchozí kapitole, tak každá skupina má lehce odlišné vlastnosti a možnosti jízdních módů a rozdělení výkonu, a proto také strategie jejich řízení se trochu liší. V rámci této práce bude návrh řídicího algoritmu pro paralelní hybrid, takže se budu věnovat převážně strategiím řízení pro tento typ hybridního vozidla, nicméně principy bývají většinou podobné pro všechny topologie. V paralelních hybridních vozidlech je z hlediska dodávky výkonu pouze jeden stupeň volnosti a tím je rozdělení točivého momentu mezi spalovacím motorem a elektromotorem, kde právě toto rozdělení je ovládáno pomocí strategie řízení.

Strategie řízení hybridního pohonu je možné rozdělit v závislosti na znalosti budoucích podmínek. V případě, že řídicí systém reaguje pouze na uplynulé a aktuální vstupní hodnoty, tak jde se o kauzální řízení. Jedná se o suboptimální řízení, protože strategie řízení nebere v potaz budoucí jízdní podmínky. To je hlavně z toho důvodu, že většinou není



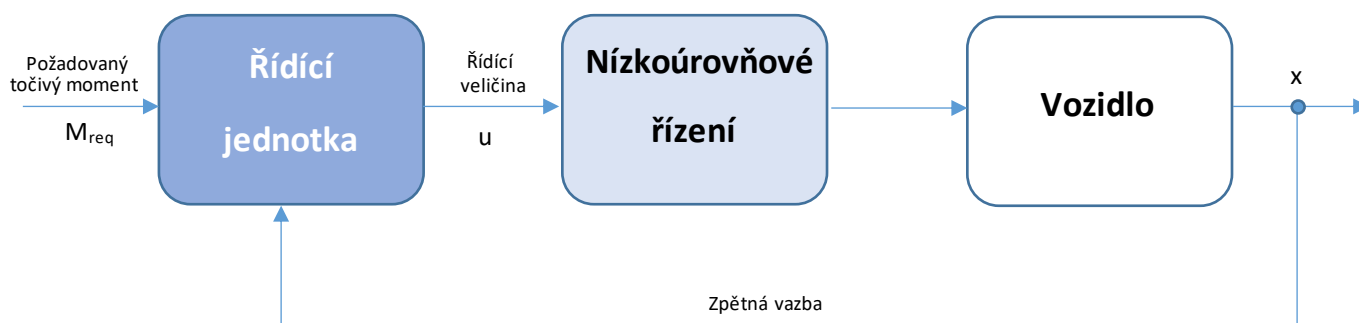
známá přesná délka trasy a jízdní profil. Tento způsob řízení je také označován jako online, protože je implementován v reálném čase. [1]

Oproti tomu v případě non-kauzálního řízení je nutné, aby byly kompletní informace o jízdním cyklu známe dopředu, tedy i budoucí jízdní podmínky. To tedy znamená, že pro optimalizaci je nutné znát rychlostní a výškový profil celé trasy. Tyto informace jsou ale dostupné většinou pouze při jízdě po standardizovaných jízdních cyklech, jako jsou například NEDC, nebo WLTC. V reálném provozu nejsou tyto informace nikdy přesně známy dopředu. Dalším problémem je vysoká výpočetní náročnost, a proto se z těchto důvodů provádí optimalizace tímto způsobem offline. Ve většině případů se potom využívá jako benchmark pro jiné, online, strategie. [1]

Dalším kritériem pro rozdělení strategií řízení je v závislosti na tom, jestli byla použita teorie optimálního řízení, nebo ne. V případě že ne, jedná se o tzv. heuristické metody. Jde o relativně jednoduché metody, které je možné snadno implementovat do řízení. Tyto strategie jsou většinou založeny na předem definovaných pravidlech, zkušenostech, nebo analýzách, podle kterých se na základě aktuálního stavu nabití baterie a požadovaného točivého momentu zvolí preferovaný pohon. Jelikož jsou tyto metody poměrně jednoduché a nejsou výpočetně náročné, tak jsou ideální pro použití při online strategiích. [1]

V případě použití teorie optimálního řízení se jedná o optimalizační metody strategie řízení. Ty jsou převážně založené na matematickém modelu systému, v tomto případě vozidla, a snaží se na určitém časovém intervalu minimalizovat spotřebu paliva za dodržení předem definovaných limit, jako například stavu nabití baterie. Tyto metody vyžadují velké množství informací o jízdním cyklu, takže jsou výpočetně náročnější než metody heuristické. Jedná se většinou o non-kauzální metody a výpočet se tedy provádí offline. Heuristické a optimalizační metody a jejich příklady budou detailněji rozebrány v dalších kapitolách. [1]

Na obrázku č. 5 je znázorněné zjednodušené blokové schéma řídicího systému hybridního vozidla. Na základě požadovaného točivého momentu rozhodne řídicí systém, v jakém poměru bude točivý moment ze spalovacího motoru a elektromotoru v každém časovém okamžiku, přičemž bere v potaz aktuální známé informace o vozidle, jako například stav nabití baterie, nebo rychlost vozidla.



Obrázek 5: Blokové schéma řídicího systému hybridního vozidla

2.1 Heuristické metody

Jak už bylo řečeno, heuristické metody jsou relativně jednoduché metody se snadnou implementací do řízení, které jsou založeny převážně na předem definovaných pravidlech nebo podmínkách, zkušenostech a analýzách. Díky své jednoduchosti, robustnosti a snadné implementaci jsou dobrými kandidáty pro online řízení, ale jelikož se nejedná o optimalizační metodu, tak se většinou nevyužije úplného potenciálu konfigurace hybridního pohonu.

Jedním ze základních principů heuristických strategií je, že spalovací motor by měl pracovat převážně v oblastech s nejvyšší účinností a v případě, že toho není možné dosáhnout, tak by se měl upřednostňovat elektrický mód pohonu a spalovací motor by měl být vypnutý. Co se týče nejvyšší účinnosti spalovacího motoru, tak ta bývá většinou při vysokém zatížení motoru. Pro dosažení vysokého zatížení spalovacího motoru se v závislosti na požadovaném výkonu motor ještě přitěžuje a nadbytečná energie se využívá na dobíjení baterie. Jelikož má pohonný systém paralelního hybridu z hlediska dodávky výkonu pouze jeden stupeň volnosti, kterým je rozdělení točivého momentu mezi spalovacím motorem a elektromotorem, musí ke změně pracovních bodů motoru dojít při aktuálních otáčkách, které jsou fixní pro rychlost vozidla a zvolený převodový poměr v daném okamžiku, na rozdíl od sériových a kombinovaných hybridů. [1] [9]

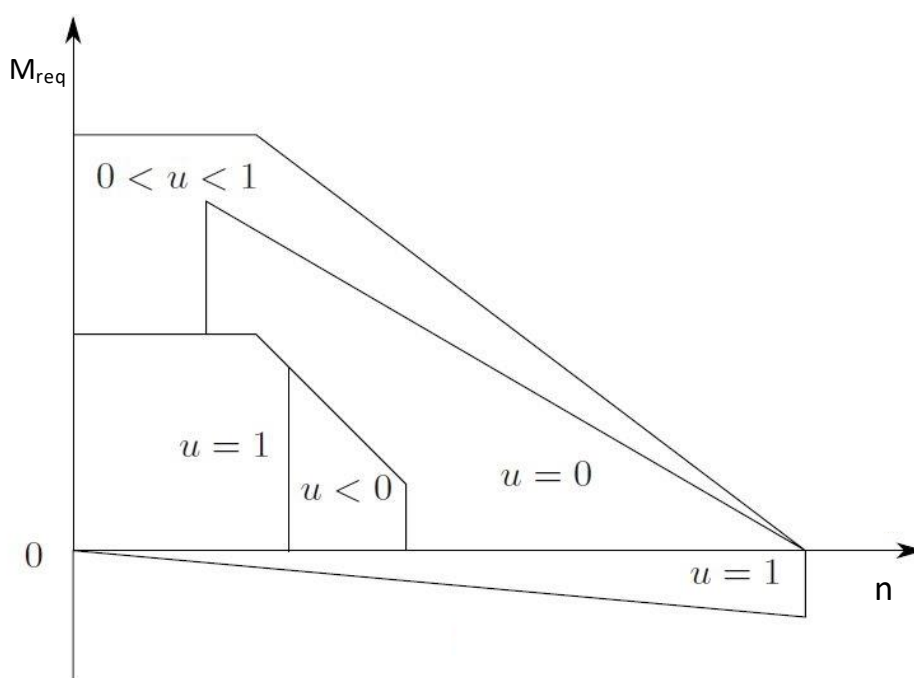
Druhým principem heuristických metod je, že stav nabití baterie by měl být neustále uvnitř předem stanovených limit a podle toho by se také mělo regulovat nabíjení a vybíjení baterie. To znamená, že když stav nabití baterie klesne pod určitou mez, mělo by se preferovat nabíjení baterie, zatímco elektrický mód je preferovanou možností, když je stav nabití baterie u horní hranice. [1] [9]



Existují dva přístupy k implementaci těchto základních principů – strategie řízení založená na předdefinovaných mapách (map-based approach) a strategie řízení založená na pravidlech nebo podmínkách (rule-based approach). Co se týká první zmíněné strategie, tak zde jsou výstupní hodnoty uloženy v předdefinovaných mapách a jejich vstupy jsou aktuální naměřené hodnoty popisující stav hnacího ústrojí. V paměti řídicí jednotky je uložena sada map. Tyto mapy se od sebe liší na základě různých parametrů hnacího ústrojí, jako například stavu nabití baterie, převodového stupně, nebo teploty motoru. Za jízdy se podle parametrů hnacího ústrojí zvolí odpovídající mapa, kde se na základě požadovaného momentu a otáček zvolí režim pohonu. [1]

Na obrázku č. 6 je pro představu ukázána mapa pro paralelní hybrid. Vstupními parametry jsou zde požadovaný točivý moment a otáčky. Na základě těchto parametrů je pak pomocí mapy zvolena hodnota u , která představuje poměr rozdělení točivého momentu mezi spalovacím motorem a elektromotorem. Veličina u může nabývat následujících hodnot:

- $u = 1$ → v provozu je pouze elektromotor, v případě záporného požadovaného momentu pracuje jako generátor
- $0 < u < 1$ → pracuje spalovací motor společně s elektromotorem
- $u = 0$ → pracuje pouze spalovací motor
- $u < 0$ → oblast střední úrovně výkonu, kde spalovací motor pomocí nadbytečné energie ještě dobíjí baterii



Obrázek 6: Strategie řízení podle předdefinovaných map [1]

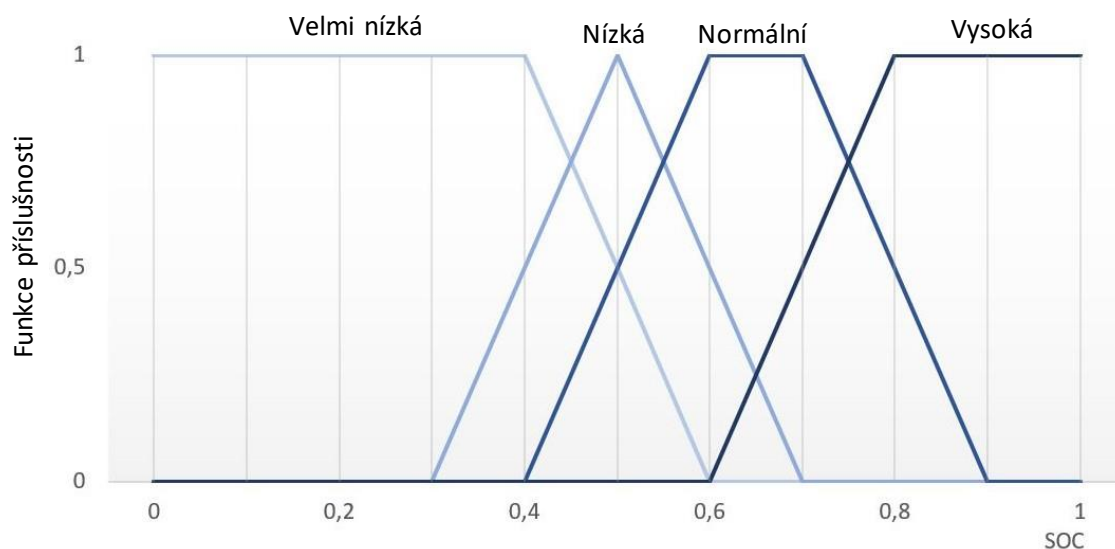


Druhým přístupem je, již zmíněná, strategie řízení založená na pravidlech (rule-based approach), která je také někdy označována pouze jako jednoduchá logika. Jednoduchá logika se skládá z booleovských výrazů – True a False. Typ jízdního módu se volí na základě toho, zdali jsou proměnné jako například stav nabití baterie, požadovaný točivý moment, nebo otáčky vyšší, nebo nižší než nějaká limitní hodnota. Pravidla jsou formulována pomocí operátorů IF – THEN (když – tak). Pro představu, k tomu, aby se zastavil spalovací motor, musí být splněny všechny následující podmínky současně:

- Aktuální rychlost vozidla a požadovaný točivý moment musí být dostatečně nízký
- Stav nabití baterie musí být dostatečně vysoký
- Příslušenství poháněné spalovacím motorem není aktivní
- Teplota spalovacího motoru, nebo katalyzátoru je dostatečně vysoká. [1]

V praxi jsou samozřejmě pravidla ještě mnohem složitější a komplexnější než v této jednoduché ukázce.

Na podobném principu, jako dva předchozí přístupy, pracuje i tzv. fuzzy logika. Fuzzy logika přináší větší flexibilitu, než booleovské výrazy True (1) a False (0), protože považuje výraz za pravdivý i když se jeho hodnota nachází v intervalu mezi 0 a 1. Pracuje s nepřesně definovanými (vágními) pojmy, jako například nízký, normální, velký nebo příliš velký. Z charakteristik baterie, elektromotoru a spalovacího motoru se stanoví funkce příslušnosti, které nemají ostré hranice. Díky tomu je pak možné v logických podmínkách pracovat i s necelými hodnotami funkce. Každá veličina má potom dvě hodnoty – fyzikální hodnotu a hodnotu funkce příslušnosti. Ukázka použití fuzzy hodnot na příkladu stavu nabití baterie (SOC) je vidět na obrázku č. 7. [1] [9] [29]



Obrázek 7: Fuzzy logika



Jednou z nevýhod heuristických metod je, že s použitím většího množství booleovských podmínek začíná být řízení velmi obtížné na ovládání a to převážně z důvodu provázanosti podmínek a určení, která z podmínek má větší prioritu. Dále je to vysoká závislost na zvolení vhodných limit, nebo map pro určení jízdních módů, což může výrazně ovlivnit jízdní podmínky. Poslední nevýhodou je již zmíněná absence optimalizace, a tudíž se ve většině případů nevyužije naplno potenciálu hybridního pohonného systému.

2.2 Optimalizační metody

Už podle názvu je zřejmé, že optimalizační metody jsou založené na teorii optimálního řízení a jsou charakterizovány matematickým modelem systému, na rozdíl od metod heuristických, které jsou založeny na pravidlech a podmínkách. „Cílem teorie optimálního řízení je určit řídicí signály, které způsobí, že proces splní fyzikální omezení a zároveň minimalizuje (nebo maximalizuje) některá výkonnostní kritéria.“ [10]

Nejdříve je vhodné vysvětlit základní problém, který řeší optimální řízení. Optimalizační metody pracují tak, že se snaží minimalizovat nějaké výkonnostní měřítko (angl. performance measure) v závislosti na řídicí hodnotě u . Při optimalizaci se použije matematický model systému, který je popsán stavovými veličinami x a dalšími parametry. Cílem je najít optimální řídicí hodnotu u^* , která minimalizuje výkonnostní měřítko, tím se dostane optimální řešení na určitém časovém intervalu. [1] [10] [11]

Výkonnostní měřítko je dáno následujícím vztahem:

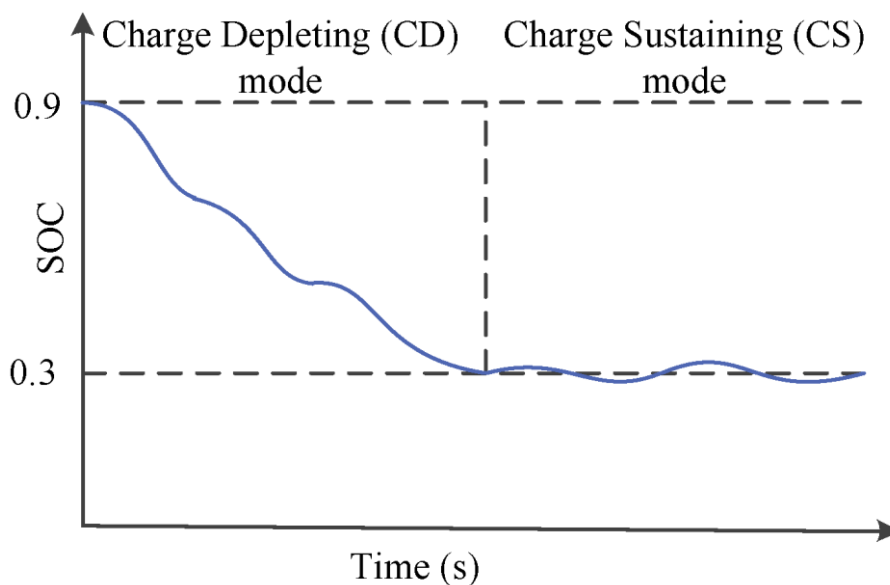
$$J_{(u)} = \int_{t_0}^{t_f} g(x(t), u(t), t) dt \quad (1)$$

Kde g představuje cenovou, nebo penalizační funkci. Tou bývá v řízení hybridních vozidel velmi často hmotnostní tok paliva \dot{m}_f , tudíž jde o minimalizaci spotřeby paliva na určitém časovém intervalu. Co se týče řídicí hodnoty u , tak ta ve většině případů představuje poměr rozdělení točivého momentu mezi spalovacím motorem a elektromotorem.

Stavové veličiny úzce souvisí s dynamikou systému, do níž jsou většinou zahrnuty ještě další subsystemy, jako například elektrické, mechanické nebo tepelné. Při optimalizačním řízení hybridních vozidel se ale pro popis systému velmi často využívají kvazistatické modely, jejichž použití počet stavových veličin výrazně sníží a zahrnují potom pouze integrální veličiny. U hybridních vozidel je stavovou veličinou nejčastěji stav nabití baterie.



V případě, že je stav nabití baterie stavovou veličinou, je nutné zavést ještě stavové omezení. Tím je, že stav nabití baterie musí zůstat uvnitř přípustných limitů po celou dobu časového intervalu. Navíc po ujetí jízdního cyklu musí stav nabití baterie nabývat takových hodnot, které jsou přípustné pro zvolený typ hybridního vozidla. S tím souvisí dva módy – charge – sustaining mode a charge – depleting mode. Micro, mild a full hybridy jsou nastaveny na charge – sustaining mode, což znamená, že koncový stav nabití baterie může mít pouze malou výchylku od stavu počátečního, takže je snaha udržet stav nabití baterie stále kolem stejné hodnoty, zatímco při charge – depleting mode může stav nabití baterie klesat. Klesat může až do nějaké minimální limitní hodnoty, které když dosáhne, tak musí nastat opět charge – sustaining mode. Tato strategie je možná ale pouze u plug-in hybridních vozidel, mimo jiné z důvodu možnosti nabití baterie z elektrické sítě. Ukázka těchto dvou módů je vidět na obrázku č. 8, kde je znázorněn průběh stavu nabití baterie v čase pro plug-in hybrid. [1] [24]



Obrázek 8: Průběh stavu nabití baterie plug-in hybridu [12]

Výsledkem optimalizace mohou být dva druhy řešení – globálně optimální řešení a lokálně optimální řešení. Cílem optimalizace není minimalizovat výkonnostní měřítko v každém časovém okamžiku, ale na určitém časovém intervalu, nebo na celém jízdním cyklu. Aby optimalizace produkovala dobré výsledky, je nutné znát informace o jízdním cyklu dopředu. Globálně optimální řešení je takové řešení, které minimalizuje výkonnostní měřítko na celé trase, tedy globálně. Tento druh řešení nastává u offline řízení, kde t_0 ve vztahu (1) je počáteční čas trasy a t_f je konečný čas trasy. Při online řízení dostáváme lokálně optimální řešení, které minimalizuje výkonnostní měřítko pouze na části trasy, tedy



lokálně. Vyjdeme-li opět ze vztahu (1), tak čas t_0 představuje aktuální čas a t_f představuje predikovaný konečný čas jízdního cyklu.

Předem ale není zřejmé, jestli optimální řešení vůbec existuje. To je proto, že může být nemožné najít řízení, které je zaprvé přípustné a zadruhé pro přípustné stavové hodnoty systému. Není moc způsobů, jak existenci ověřit, a proto je ve většině případů výhodnější snažit se najít optimální řešení, než se snažit ověřit jeho existenci. [10]

Optimalizační úlohy je možné řešit pomocí množství různých algoritmů. V následující sekci budou přiblíženy některé algoritmy, které se při řízení hybridního pohonného systému používají nejčastěji.

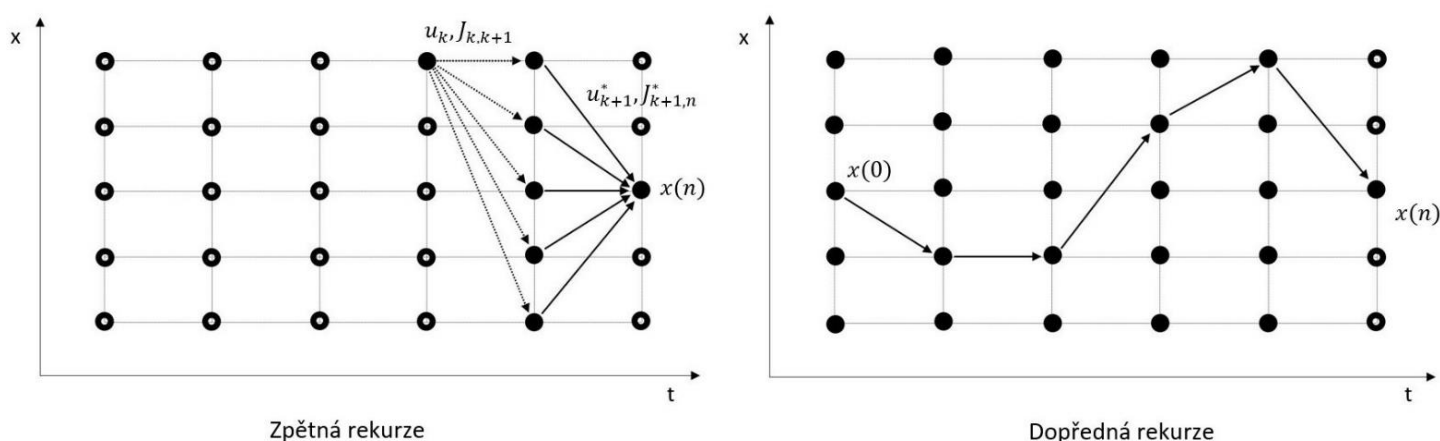
2.2.1 Dynamické programování

Prvním algoritmem, který se velmi často používá pro řešení optimalizačních úloh řízení hybridního pohonného systému je dynamické programování. Dynamické programování představuje offline optimalizační metodu, tudíž jízdní cyklus musí být dopředu známý. Hlavní výhodou tohoto algoritmu je, že zaručuje globálně optimální řešení optimalizačního problému. Další výhodnou vlastností této metody je, že výpočetní zátěž se zvyšuje pouze lineárně s konečným časem cyklu. I přesto je ale značnou nevýhodou vysoká výpočetní náročnost, která navíc ještě exponenciálně roste s počtem stavových hodnot systému. Proto se dynamické programování používá převážně na jednodušší systémy a není vhodné do reálného provozu. Velmi často se využívá jako benchmark pro jiné, ve většině případů online, optimalizační metody. Vývoj a optimalizace algoritmu dynamického programování jsou jedny ze stěžejních částí této práce a proto budou v této kapitole vysvětleny převážně základní principy této metody a pak samotná implementace do strategie řízení hybridních vozidel bude ve větších detailech popsána až v následujících kapitolách. [1] [10]

Dynamické programování je založeno na dodržování principu optimality při víceúrovňových rozhodovacích procesech. Princip optimality, jehož autorem je Richard Ernest Bellman, říká, že „*optimální strategie má takovou vlastnost, že ať je počáteční stav a počáteční rozhodnutí jakékoli, zbývající rozhodnutí musí tvořit optimální strategii pro stav, který je výsledkem prvního rozhodnutí.*“ [13] Díky této vlastnosti je při výpočtu mnohem výhodnější počítat od konečného stavu a jít zpět, než od začátku a počítat všechny možné stavy systému. Při postupu odzadu se v každém kroku vypočítá optimální řízení od aktuálního stavu do konce a tím se celkový počet výpočtů razantně sníží. Takovému postupu se říká zpětná rekurse.



Na obrázku č. 9 je algoritmus dynamického programování schematicky znázorněn. Při řízení hybridního vozidla se v každém okamžiku jízdního cyklu určuje optimální rozdělení točivého momentu s tím, aby se dosáhlo předem definované hodnoty stavové veličiny na konci cyklu, kterou tedy v tomto případě většinou bývá stav nabití baterie. Každé toto rozhodnutí s sebou nese cenovou funkci, která souvisí se změnou stavu z aktuálního na následující. Optimální řízení a optimální cenová funkce této následující fáze byla vypočítána v předchozím kroku. Optimální řízení pro aktuální fázi je takové řízení, které má nejmenší součet cenové funkce aktuálního stavu a optimální cenové funkce pro dosažení konečného stavu z následující fáze. [1] [10] [25]



Obrázek 9: Schématické znázornění procesu dynamického programování

Co se týče samotného výpočtu algoritmu, je nejdříve nutné provést diskretizaci času a stavových hodnot systému. Díky tomu dostaneme matici $n \times m$, kde n je diskretizovaný čas jízdního cyklu a m představuje diskretizaci stavu systému. Pro výpočet je potřeba stanovit počáteční podmínku v podobě konečného stavu systému $x(n)$, který musí být na konci jízdního cyklu dosažen. Poté již může začít výpočet, kde pro každý bod matice se vypočítá řídicí hodnota u a její cenová funkce. Jako řídicí hodnota se u hybridních vozidel velmi často používá točivý moment elektromotoru a je opět nutná diskretizace. Optimální řídicí hodnota u^* v každém bodě je řízení s nejnižší hodnotou součtu cenové funkce $J_{k, k+1}$ pro dosažení následující fáze $k+1$ a optimální ceny $J_{k+1, n}^*$ spojené s dosažením z bodu následující fáze $k+1$ do finální fáze. Tomuto procesu se říká zpětná rekurze, kterou je možné matematicky popsat následujícím vztahem:

$$u_k^*(k) = \arg(J_{k, n}^*) = \arg\left(\min_u \left(C_{k, n}(x_k^i, u_k^j)\right)\right) = \arg\left(\min_u \left(J_{k, k+1}(x_k^i, u_k^j) + J_{k+1, n}^*(x_{k+1}^i)\right)\right) \quad (2)$$



Výsledkem je stavově-časová matice, která v každém bodě poskytuje informaci o optimálním řízení a minimální ceně. Pak může proběhnout dopředná rekurze. Výpočet dopředné rekurze probíhá od počátku jízdního cyklu, případně ze zvolené fáze k , za stavu x . Stavově-časová matice pak slouží v podstatě jako look-up table pro interpolaci trajektorie optimálního stavu x^* podle optimálního řízení vypočítaného při zpětné rekurzi. [10] [26] [27]

2.2.2 Pontrjaginův princip minima

Další významnější metodou pro řešení optimalizačního problému je Pontrjaginův princip minima. Tato metoda je založená na variačním počtu a cílem je najít globální extrém funkcionálu. Funkcionálem je výkonnostní měřítko a opět je snahou získat jeho minimální hodnotu pro získání optimálního řízení. K výpočtu se používá tzv. Hamiltonián, který je dán následujícím vztahem:

$$\mathcal{H}(x(t), u(t), p(t), t) \triangleq g(x(t), u(t), t) + p^T(t)[a(x(t), u(t), t)] \quad (3)$$

Kde p jsou Lagrangeovy multiplikátory a g představuje cenovou, nebo penalizační funkci ze vztahu (1). [10] [28]

Pontrjaginův princip minima říká, že optimální řízení musí minimalizovat Hamiltonián a tudíž musí platit následující vztah:

$$\mathcal{H}(x^*(t), u^*(t), p^*(t), t) \leq \mathcal{H}(x^*(t), u(t), p^*(t), t), \text{ pro } t \in \langle t_0, t_f \rangle \quad (4)$$

Aby bylo u^* optimálním řízením systému, musí pro čas $t \in \langle t_0, t_f \rangle$ a všechny přípustné řídicí hodnoty platit následující podmínky:

$$\dot{x}(t) = \frac{\partial \mathcal{H}}{\partial p}(x^*(t), u^*(t), p^*(t), t) \quad (5)$$

$$\dot{p}^*(t) = -\frac{\partial \mathcal{H}}{\partial x}(x^*(t), u^*(t), p^*(t), t) \quad (6)$$

$$\mathcal{H}(x^*(t), u^*(t), p^*(t), t) \leq \mathcal{H}(x^*(t), u(t), p^*(t), t) \quad (7)$$

Ve srovnání s algoritmem dynamického programování je tato metoda méně výpočetně náročná a lze ji při mírném zjednodušení použít v reálném provozu jako online strategii řízení. Na tomto principu pracuje také například metoda minimalizace ekvivalentu spotřeby paliva. [10] [28]



2.3 Prediktivní a adaptivní strategie řízení

Ještě jednou z možností rozdělení strategií řízení hybridního pohonu je na prediktivní a adaptivní strategii řízení. Prediktivní a adaptivní strategie lze použít pro všechny typy pohonných systémů, ale největší uplatnění mají tyto metody u hybridních vozidel a jejich význam roste s mírou elektrifikace pohonu. K zajištění požadovaných funkcí těchto metod je nutné získávat informace od jiných systémů a senzorů ve vozidle, případně i informace od jiných vozidel, nebo z infrastruktury. Informace se získávají například pomocí navigace, car-to-car komunikace, car-to-infrastructure komunikace, nebo ze statistik a podle stylu jízdy řidiče.

Prediktivní strategie řízení

Prediktivní strategie potřebují ke své činnosti znát informace o celém jízdním cyklu, případně jeho části, stylu jízdy řidiče a provozu. V první fázi pracuje strategie obdobně jako offline metody řízení – před začátkem jízdy proběhne výpočet celé trasy a odsimuluje se celý jízdní cyklus pro nalezení optimálního průběhu a optimální rychlosti, kde hlavním požadavkem je dojet do cíle v požadovaném čase, s minimální spotřebou energie a minimálními emisemi. Dále řešení probíhá v časových krocích, kde se na základě rychlostního, výškového a směrového profilu stanoví potřebný výkon, nebo točivý moment. V každém kroku se určí optimální rozdělení točivého momentu mezi spalovacím motorem a elektromotorem a případně se zvolí ještě preferovaný převodový stupeň. To všechno probíhá s respektováním požadovaného času dojezdu do cílové destinace. Výpočet se většinou provádí mimo vozidlo na výkonném serveru v cloudu. Během samotné jízdy pak probíhá průběžná kontrola času a spotřebované energie. [29]

Mezi hlavní funkce strategie patří optimalizace spotřeby energie, u plug-in hybridů s preferencí využití elektrického pohonu, posílení battery management, aby například nedocházelo k nadměrnému vybíjení baterie, nebo dodržování požadovaného jízdního času. K tomu slouží různé prostředky řízení. Některé z nich jsou uvedeny v následujícím seznamu:

- Informování řidiče o ekonomice jízdy, preferované rychlosti, nebo volbě převodového stupně
- Přizpůsobení stavu nabití baterie – navýšení, nebo snížení v závislosti na předpokládaných podmínkách
- Jízda setrvačností před místem s omezenou rychlostí
- Na základě informací o dopravě zvolit vhodnou trasu. [29]



Adaptivní strategie řízení

Druhou možnou variantou je adaptivní strategie řízení. Jak už může naznačovat samotný název, tak základní vlastností této metody je to, že se sleduje okolí vozidla a strategie adaptuje jízdu aktuálním podmínkám v reálném čase. K tomu je nutné, aby bylo vozidlo vybavené pokročilými jízdními asistenty a senzory, které neustále sledují okolí vozidla, a s tím spojený software pro zpracování těchto informací. Dále vozidla disponují vyspělými komunikačními systémy, jako například již zmiňované car-to-car komunikace, pro získávání informací z okolních vozidel, a car-to-infrastructure komunikace, pro získání informací z infrastruktury. Tyto informace slouží mimo ekonomii provozu také pro zvýšení bezpečnosti provozu a mohou být využívány i pro autonomní jízdu vozidla. [29]



3 Vývoj řídicího algoritmu

Hlavním cílem této práce je vývoj a optimalizace globálně optimálního algoritmu pro řízení hybridního pohonu s využitím dynamického programování. Tato následující kapitola se bude zabývat návrhem tohoto algoritmu. Postupně bude popsán postup návrhu matematického modelu pohonného systému hybridního vozidla, výpočet parametrů nezbytných pro řízení a samotný řídicí algoritmus využívající metodu dynamického programování.

Jak již bylo řečeno, tak globálně optimální řešení je takové řešení, které minimalizuje výkonnostní měřítko na celé trase, tedy globálně. Z tohoto důvodu je pro fungování algoritmu nutné znát informace o jízdním cyklu předem a jedná se tedy o offline strategii řízení hybridního pohonu, která se ve většině případů používá jako benchmark pro online řídicí strategie. Pro výpočet optimální trajektorie stavu nabití baterie proto musí být známý rychlostní a výškový profil celé trasy, nebo jízdního cyklu. Na základě těchto vstupních parametrů se v každém časovém kroku pro matematický model hybridního vozidla vypočítá požadovaný točivý moment. Úkolem algoritmu je pak optimálně rozdělit tento točivý moment mezi spalovací motor a elektromotor tak, aby se dosáhlo minimální spotřeby energie a minimálních emisí.

Co se týče matematického modelu vozidla a jízdního cyklu, tak s nimi přicházejí do algoritmu drobné nepřesnosti, nebo problém s výpočetní náročností. I když známe kompletní informace o jízdním cyklu, tak predikovaný průběh stavu nabití baterie nebude optimální v reálném životě, například z důvodu náhodné změny provozu, která ovlivní rychlostní profil jízdního cyklu. U modelu hybridního vozidla je zásadní problém v tom, že je nutné najít nějaký kompromis mezi zjednodušením systému a vysokou výpočetní náročností. Výpočet optimálního řízení metodami jako je dynamické programování pro komplexní modely může trvat až několik hodin, proto je největší výzvou této práce vytvořit efektivnější výpočetní algoritmus pro zjednodušený model hybridního vozidla, který by dokázal provést výpočet s dostatečnou přesností v rámci desítek sekund, nebo maximálně několika minut.

Celý řídicí algoritmus je tvořen v programovacím jazyku Python. Python byl zvolen z důvodu relativně jednoduchého používání, takže je ideální i pro začátečníky, ale zároveň se jedná o výkonné řešení, které je velmi rozšířené napříč všemi odvětvími technického průmyslu.



3.1 Model vozidla

Optimalizační strategie řízení jsou postaveny na matematickém modelu, který popisuje systém. Aby se při řízení dosáhlo optimálního rozdělení točivého momentu mezi spalovacím motorem a elektromotorem, je nutné nejdříve v první fázi vytvořit model vozidla. Tento model v průběhu jízdního cyklu reflektuje dynamiku systému a získávají se díky němu důležité parametry nezbytné pro optimalizaci. Pro účely této práce je nezbytné určité parametry modelu zjednodušit, nebo zanedbat, protože použitím komplexního modelu hybridního vozidla by se několikanásobně zvýšila výpočetní náročnost algoritmu. Zjednodušení představuje například nahrazení některých veličin, jejichž hodnota se lehce mění v závislosti na jiných parametrech, konstantami (např. koeficient odporu valení).

Co se týče tvorby modelu hybridního vozidla, tak existují dva přístupy – kvazistatický přístup a dynamický přístup. V následující části budou popsány oba přístupy modelování a také zdůvodněno, který z nich je preferovaný k použití pro řídicí algoritmus hybridního pohonného systému.

Kvazistatický přístup

Prvním přístupem k modelování je kvazistatický přístup a jedná se o jednodušší z těchto dvou metod. Vstupními hodnotami z jízdního cyklu jsou rychlost, zrychlení a úhel stoupání vozovky. Výstupní hodnotou je pak hnací síla na kolech, která se dopočítá pomocí vstupních hodnot a dalších parametrů vozidla. Tyto výpočty se provádějí za předpokladu, že rychlost, zrychlení a úhel stoupání jsou konstantní pro dostatečně krátký časový interval, aby splnil tento předpoklad. Celý jízdní cyklus je tedy rozdělen na tyto intervaly a jejich délka bývá ve většině případů 1 sekunda, ale může nastat i případ, kdy časové intervaly nejsou konstantní.

Kvazistatický přístup modelování je velmi výhodný pro minimalizaci spotřeby energie i u komplexních systémů. Díky tomu se využívá u řídicích systémů pro určení rozdělení toku výkonu a predikce celkové ekonomie provozu dosahuje velmi dobré přesnosti. Výpočetní náročnost i přes tyto schopnosti je relativně nízká. Jednou z nevýhod tohoto přístupu je fakt, že model musí následovat jízdní cyklus a ten musí být dopředu známý. Proto tento přístup není možné použít například k řešení problémů se zpětnou vazbou. [1]

Na základě zmíněných výhod a jednoduchosti oproti dynamickému přístupu bude při návrhu algoritmu v této práci používán kvazistatický přístup modelování.



Dynamický přístup

Dynamický přístup je založen na přesném matematickém popisu systému. Když se zaměříme na model pohonného systému hybridního vozidla, tak je model ve většině případů tvořen soustavou obyčejných diferenciálních rovnic, které mají formu jako následující vztah:

$$\frac{d}{dt}x(t) = f(x(t), u(t)) \quad (8)$$

Pomocí tohoto vztahu je možné popsat velké množství dynamických účinků pohonného systému a některé optimalizační úlohy lze řešit pouze s použitím dynamického přístupu. Mezi ně patří například úlohy řízení se zpětnou vazbou. Zásadní nevýhodou tohoto přístupu je velmi vysoká výpočetní náročnost. [1]

3.1.1 Parametry vozidla a jeho konfigurace

Jako model pro řídicí algoritmus byl zvolen paralelní hybrid v konfiguraci P2, což znamená že elektromotor je umístěn mezi spalovacím motorem a převodovkou. Tento typ konfigurace byl zvolen z toho důvodu, že se jedná o nejběžnější typ paralelního hybridu. Důležité parametry vozidla jsou vypsány v tabulce č. 1.

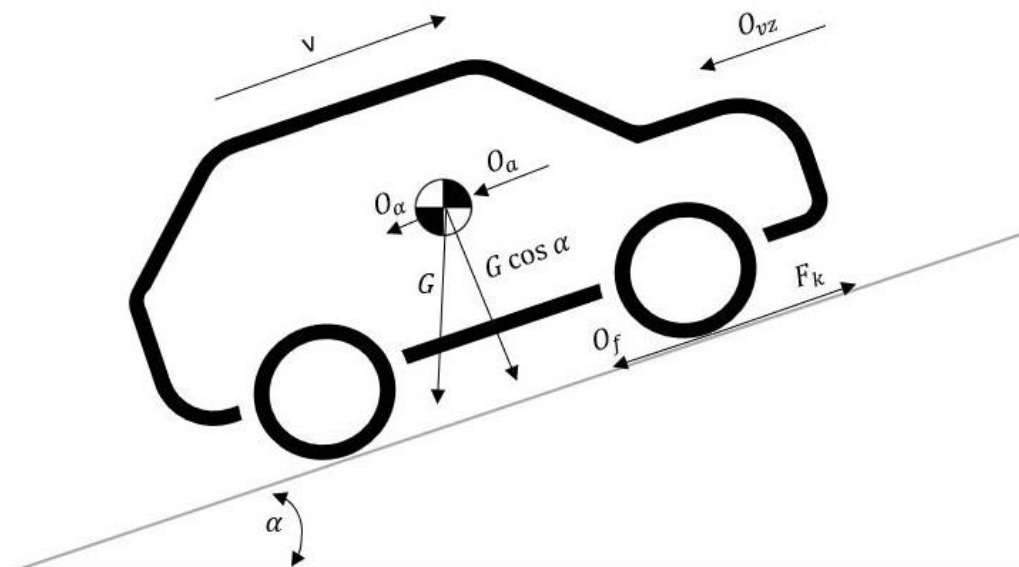
Hmotnost vozidla	$m = 1750 \text{ kg}$
Spalovací motor	Zážehový motor 1.3 l, 96 kW
Výkon elektromotoru	$P_{EM} = 65 \text{ kW}$
Kapacita baterie	$C_{Bat} = 14 \text{ kWh}$
Napětí baterie	$U_{Bat} = 380 \text{ V}$
Dynamický poloměr pneumatiky	$r_d = 0,3155 \text{ m}$
Stálý převod diferenciálu	$i_{dif} = 3,56$
Převodové poměry	$i_g = [3.78, 2.05, 1.28, 0.94, 0.78, 0.67]$
Účinnost převodového ústrojí	$\eta_g = 0,95$
Průmět čelní plochy vozidla	$S_x = 2 \text{ m}^2$
Součinitel odporu vzduchu	$c_x = 0,28$
Součinitel odporu valení	$f = 0,019$
Součinitel rotačních hmot	$\delta = 1,15$

Tabulka 1: Vybrané parametry vozidla



3.1.2 Výpočet požadované trakční síly

Po zavedení konfigurace a základních parametrů vozidla je další fází výpočet požadované trakční síly. V první řadě je nutné zjistit dynamické chování vozidla. K popisu pohybu slouží model podélné dynamiky. Vozidlo je přitom považováno za hmotný bod, na který působí síly v podélném směru. Schéma podélné dynamiky vozidla je znázorněná na obrázku č. 10.



Obrázek 10: Schéma podélné dynamiky vozidla

Kde F_k je trakční síla na kolech, G představuje tíhu vozidla, α je úhel sklonu vozovky, O_f je odpor valení kol, O_{vz} je odpor vzduchu, O_α je odpor stoupání a O_a je odpor ze zrychlení vozidla.

Na základě modelu podélné dynamiky dostaneme silovou rovnováhu ve směru jízdy a získáme následující vztahy:

$$m_{red} \cdot \dot{v}(t) = F_k(t) - O_f(t) - O_\alpha(t) - O_{vz}(v(t)) \quad (9)$$

$$m_{red} = m \cdot \delta \quad (10)$$

Kde m_{red} je redukovaná hmotnost vozidla, která bere v potaz i setrvačnost rotujících součástí pohonného systému, a je dána součinem hmotnosti vozidla m a součinitelem rotačních hmot δ . $\dot{v}(t)$ je zrychlení vozidla v závislosti na čase. Ze vztahu (9) je tedy zřejmé, že k výpočtu je nutné určit jízdní odpory vozidla.



Odpor valení

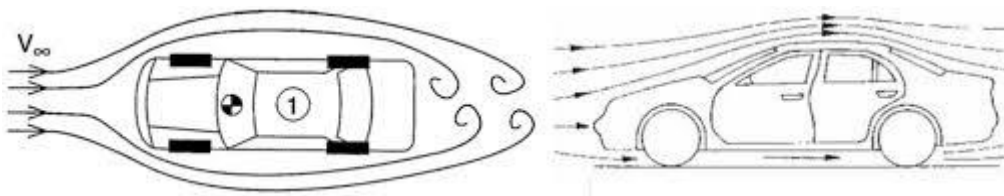
Valivý odpor vzniká, když se těleso kruhového průřezu odvaluje po pevné podložce, v případě vozidla je to tedy pneumatika. Pneumatika se pod tíhou vozidla deformuje a přenáší síly z vozidla na vozovku a naopak. Mají-li všechna kola na vozidle stejnou hodnotu součinitele valivého odporu, tak platí tento vztah:

$$O_f(t) = m \cdot g \cdot f \cdot \cos \alpha(t) \quad (11)$$

Kde m je hmotnost vozidla, g je gravitační zrychlení a f je součinitel odporu valení. Součinitel odporu valení ve skutečnosti není konstantní, je závislý například na povrchu a stavu vozovky, nebo rychlosti jízdy, ale z důvodu zjednodušení modelu bude v této práci brán za konstantu.

Odpor vzduchu

Odpor vzduchu je způsoben prostředím, ve kterém se vozidlo pohybuje. Vyjadřuje odporovou sílu laminárního a turbulentního obtékání karoserie, která směřuje proti pohybu vozidla.



Obrázek 11: Aerodynamika vozidla [17]

Pro výpočet odporu vzduchu platí vztah:

$$O_{vz}(t) = \frac{1}{2} \cdot \rho_{vz} \cdot S_x \cdot c_x \cdot v(t)^2 \quad (12)$$

Kde ρ_{vz} je hustota vzduchu, která se za běžných podmínek udává jako $\rho_{vz} = 1,25 \text{ kg/m}^3$. S_x je čelní plocha vozidla a c_x je součinitel odporu vzduchu.

Odpor stoupání

Dalším jízdním odporem, který ve vztahu (9) figuruje, je odpor ze stoupání. Ten vzniká při jízdě vozidla na podélném svahu. Pro výpočet odporu stoupání platí následující vztah:

$$O_\alpha(t) = m \cdot g \cdot \sin \alpha(t) \quad (13)$$

¹ Simulace v rámci této práce se provádí pro homologační cykly bez stoupání, a proto je odpor stoupání ve výpočetním algoritmu v programu Python zanedbán.



Odpor ze zrychlení

Poslední z jízdních odporů, který figuruje ve vztahu (9) je odpor ze zrychlení, který je někdy označován i jako setrvačný odpor. Vzniká při změně rychlosti vozidla tak, že při akceleraci působí odporová síla proti směru jízdy. Když vozidlo zpomaluje působí síla naopak ve směru pohybu vozidla. Odpor tvoří dvě složky – odpor posuvných částí a odpor zrychlení rotačních částí. Výpočet je naznačen ve vztahu č. 14:

$$O_a(t) = m \cdot \delta \cdot a(t) \quad (14)$$

Kde δ je součinitel rotačních hmot, a a je zrychlení vozidla. Co se týče součinitele rotačních hmot, tak znovu platí, že se reálně nejedná o konstantu, protože závisí například na převodovém stupni, ale pro účely této práce je hodnota součinitele opět zvolena jako konstantní.

Hodnota zrychlení není přímo vstupní hodnotou získávanou z jízdního cyklu, ale jelikož je jízdní cyklus rozdělený na množství časových intervalů, je možné zrychlení i tak určit. Vstupní hodnotou z jízdního cyklu je okamžitá rychlost vozidla zaznamenávána za určitý časový interval, a proto je možné vyjádřit zrychlení ze vztahu (14) jako změnu rychlosti za časový interval. Pak můžeme vztah přepsat v následující formě:

$$O_a(t) = m \cdot \delta \cdot \frac{\Delta v(t)}{\Delta t} \quad (15)$$

Dosazením vztahů (11), (12), (13) a (15) do silové rovnováhy získáváme následující vztah:

$$m \cdot \delta \cdot \frac{\Delta v(t)}{\Delta t} = F_k(t) - m \cdot g \cdot f \cdot \cos \alpha(t) - m \cdot g \cdot \sin \alpha(t) - \frac{1}{2} \cdot \rho_{vz} \cdot S_x \cdot c_x \cdot v(t)^2 \quad (16)$$

A následně upravením této rovnice dostaneme vztah pro výpočet trakční síly vozidla, který vypadá takto:

$$F_k(t) = m \cdot g \cdot (f \cdot \cos \alpha(t) + \sin \alpha(t)) + \frac{1}{2} \cdot \rho_{vz} \cdot S_x \cdot c_x \cdot v(t)^2 + m \cdot \delta \cdot \frac{\Delta v(t)}{\Delta t} \quad (17)$$



3.1.3 Výpočet požadovaného točivého momentu

Na základě zjištění požadované trakční síly na kolech je možné pokračovat v modelování pohonného systému vozidla. Pro algoritmus řízení hybridního pohonu je nutné získat požadovaný točivý moment v každém časovém intervalu. Požadovaný točivý moment je součtem momentů spalovacího motoru a elektromotoru, které jsou zdrojem trakční síly, a proto musí platit následující vztah:

$$F_k(t) = (M_{SM}(t) + M_{EM}(t)) \cdot i_g(t) \cdot i_{dif} \cdot \eta_g \cdot r_d \quad (18)$$

Kde M_{SM} je točivý moment spalovacího motoru, M_{EM} je točivý moment elektromotoru, i_g je zvolený převodový poměr, i_{dif} je stálý převod diferenciálu, η_g je společná mechanická účinnost převodovky a diferenciálu a r_d je dynamický poloměr pneumatiky.

Jelikož se jedná o paralelní hybrid konfigurace P2, musí být otáčky a úhlová rychlost spalovacího motoru a elektromotoru stejná, a proto platí tento vztah:

$$\omega_{SM}(t) = \omega_{EM}(t) = \omega(t) \quad (19)$$

Kde ω_{SM} je úhlová rychlost spalovacího motoru, ω_{EM} je úhlová rychlost elektromotoru a ω je úhlová rychlost na vstupní hřídeli převodovky, která se získá výpočtem z okamžité rychlosti vozidla pomocí následujícího vztahu:

$$\omega(t) = \frac{v(t) \cdot i_g(t) \cdot i_{dif}}{r_d} \quad (20)$$

Otáčky na vstupní hřídeli převodovky se určí tímto způsobem:

$$n(t) = \frac{30 \cdot \omega(t)}{\pi} \quad (21)$$

Požadovaný točivý moment, jehož zdrojem je spalovací motor a elektromotor, získáme pomocí vztahů (17) a (18) a po úpravě dostáváme následující vztah:

$$M_{req}(t) = (M_{SM}(t) + M_{EM}(t)) = \frac{F_k(t)}{i_g(t) \cdot i_{dif} \cdot \eta_g \cdot r_d} \quad (22)$$

Pro další výpočty je důležitý ještě požadovaný výkon. Ten je možné vypočítat podle vztahu (23):

$$P_{req}(t) = \frac{F_k(t) \cdot v(t)}{\eta_g} \quad (23)$$



3.1.4 Baterie

Elektrochemické baterie jsou jednou z nejdůležitějších součástí pohonného systému hybridních vozidel. Baterie je zařízení, které transformuje chemickou energii na elektrickou a naopak, jehož hlavní funkcí je úložiště elektrické energie.

V hybridních a v čistě elektrických vozidlech jsou zdrojem elektrické energie trakční baterie. Trakční baterie jsou charakterizovány pomocí dvou základních parametrů – výkonem a kapacitou baterie. Co se týče výkonu, tak ten představuje primární charakteristiku baterie a musí splnit výkonové požadavky elektrických zařízení v pohonném systému. Kapacita baterie vyjádřená jako integrál elektrického proudu, který může být dodán plně nabitou baterií, při jejím kompletní vybití. Kapacita baterie by měla splnit předepsané dojezdové specifikace a bývá udávána většinou v *kWh* (kilowatt hodinách), nebo v *Ah* (ampér hodinách). Existuje ještě bezrozměrná veličina, kterou je stav nabití baterie, anglickou zkratkou *SOC*, která je velmi důležitá pro řídicí algoritmus. Stav nabití baterie představuje procentuální vyjádření zbylé kapacity baterie oproti jmenovité kapacitě baterie. [1]

Co se týče baterií pro hybridní a elektrická vozidla, tak ty mají několik požadovaných vlastností. Jsou jimi například vysoký měrný výkon, vysoká měrná energie, dlouhá životnost, vysoká spolehlivost a nízké pořizovací a servisní náklady. Pro hybridní vozidla je asi nejdůležitějším parametrem měrný výkon. Měrný výkon určuje zrychlení a maximální rychlost, jaké je vozidlo schopné dosáhnout. Jednotkou měrného výkonu baterie je *W/kg*. Dalším důležitým parametrem je měrná energie. Měrná energie představuje množství energie, která může být uložena v baterii, na jednotku hmotnosti. Obvykle je dána jednotkou *Wh/kg*. Tento parametr bere v potaz to, že celá kapacita baterie nemůže být ve skutečnosti využita. To je z toho důvodu, že baterie má vždy stanovené maximální a minimální limitní hodnoty pro stav nabití baterie, které nemohou být při nabíjení a vybíjení překročeny. [1]

Baterie používané v automobilovém průmyslu musí být vždy dobíjitelné. V současnosti jsou ve vozidlech nejvíce používány následující druhy baterií:

- Olovo – kyselinové baterie
- Nikl – metal hydridové baterie
- Baterie na bázi lithia (například lithium-ion)

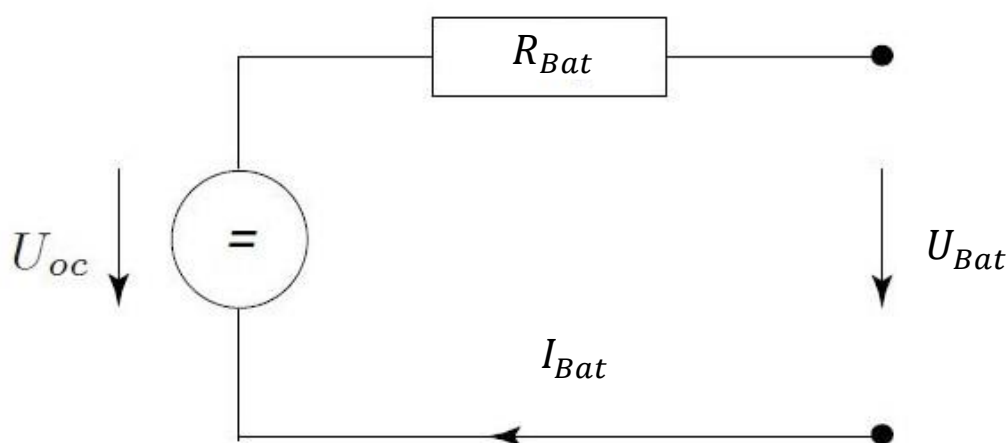


Kvazistatický model baterie

Nejdůležitějším parametrem samotného modelování baterie je stav nabití baterie. Stav nabití baterie je funkcí protékajícího proudu. Základní vztah pro změnu stavu nabití baterie za časový krok je dán následující rovnicí:

$$\Delta SOC(t) = -\Delta t \frac{I_{Bat}(t)}{C_{Bat}} \quad (24)$$

Kde I_{Bat} je elektrický proud proudící z, nebo do baterie. V případě vybíjení baterie se bere kladná hodnota elektrického proudu a při nabíjení baterie se bere hodnota záporná. C_{Bat} je kapacita baterie.



Obrázek 12: Náhradní schéma baterie [1]

Pro výpočet stavu nabití baterie je tedy nutné stanovit elektrický proud baterie. Ten lze vypočítat na základě náhradního schématu baterie z obrázku č. 12 a pak platí následující rovnice:

$$I_{Bat}(t) = \frac{U_{OC}(t) - \sqrt{U_{OC}^2(t) - 4 \cdot R_{Bat}(t) \cdot P_{Bat}(t)}}{2 \cdot R_{Bat}(t)} \quad (25)$$

Kde U_{OC} je otevřený okruh napětí, R_{Bat} je vnitřní odpor baterie a P_{Bat} je výkon baterie, který baterie buď dodává, nebo odebírá. Aby byla splněna podmínka reálného kořenu, musí být hodnota pod odmocninou větší než nula. Kvůli tomu jsou zavedeny limity pro výkon baterie:

$$P_{Bat,min}(t) \leq P_{Bat}(t) \leq P_{Bat,max}(t) \quad (26)$$



Kombinací rovnic (24) a (25) získáváme vztah pro výpočet změny stavu nabití baterie za časový interval Δt , který je dán následující rovnicí:

$$\Delta SOC(t) = -\Delta t \frac{U_{OC}(t) - \sqrt{U_{OC}^2(t) - 4 \cdot R_{Bat}(t) \cdot P_{Bat}(t)}}{2 \cdot R_{Bat}(t) \cdot C_{Bat}} \quad (27)$$

Hodnota ΔSOC nabývá záporných hodnot při vybíjení baterie, zatímco při nabíjení baterie nabývá hodnot kladných. Stav nabití baterie v následujícím kroku $t+1$ je pak možné získat pomocí těchto rovnic:

$$SOC(t + 1) = SOC(t) - \Delta t \frac{I_{Bat}(t)}{C_{Bat}} \quad (28)$$

$$SOC(t + 1) = SOC(t) - \Delta t \frac{U_{OC}(t) - \sqrt{U_{OC}^2(t) - 4 \cdot R_{Bat}(t) \cdot P_{Bat}(t)}}{2 \cdot R_{Bat}(t) \cdot C_{Bat}} \quad (29)$$

3.1.5 Elektromotor

Elektrické stroje jsou nezbytnou součástí každého vozidla. U konvenčních vozidel mají elektrické stroje funkci startéru, nebo alternátoru. Pomocí startéru dosáhne spalovací motor volnoběžných otáček a může začít produkovat točivý moment, zatímco alternátor slouží k produkci elektřiny pro dobíjení baterie a elektrického příslušenství. U hybridních a elektrických vozidel jsou ale elektrické stroje nezbytnou součástí a mají využití hned několika způsoby. V první řadě jako trakční elektromotor, který přeměňuje elektrickou energii z baterie na mechanickou k pohonu vozidla. A potom je to jako generátor, který naopak přeměňuje mechanickou energii na elektrickou pro dobíjení baterie. V případě rekuperace při brzdění slouží generátor opět k transformaci mechanické energie na energii elektrickou. U paralelních hybridů obě tyto funkce zastane jeden elektrický stroj, zatímco u sériových a kombinovaných hybridů musí být elektrické stroje dva – trakční motor a menší generátor. [1]

U hybridních vozidel jsou nejdůležitějšími vlastnostmi elektromotoru například vysoký měrný výkon, vysoká účinnost, nízká pořizovací cena, nízká hlučnost nebo dobrá ovladatelnost.

Elektromotory je v podstatě možné rozdělit do dvou skupin – stejnosměrné elektromotory a motory na střídavý proud. V automobilovém průmyslu jsou ale nejvíce zastoupeny elektromotory na střídavý elektrický proud a mezi nejpoužívanější druhy patří asynchronní motory a synchronní motory s permanentními magnety.



Asynchronní motor pro svoji činnost využívá elektromagnetické indukce, a proto je také označován jako indukční motor. „Základem činnosti asynchronního stroje je vytvoření točivého magnetického pole statoru, které vznikne průchodem střídavého trojfázového proudu vinutím statoru. Toto magnetické pole indukuje v rotoru napětí a vzniklý proud rotoru vyvolává magnetický tok, který je spřažen se statorem. Spřažený magnetický tok vyvolá silové působení na rotor a tím otáčení rotoru. Asynchronní stroj může dávat na výstupní hřídeli kroutící moment jen tehdy, pokud rychlost otáčení magnetického pole statoru je rozdílná oproti mechanickým otáčkám rotoru.“ [18]

U synchronního motoru s permanentními magnety je střídavý proud přiveden na stator, kde vytváří točivé magnetické pole. Rotor obsahuje permanentní magnety, které vytvářejí konstantní magnetické pole a práce motoru je založena na interakci točivého magnetického pole na statoru a konstantního magnetického pole na rotoru. [19]

Kvazistatický model elektromotoru

Při tvorbě kvazistatického modelu elektromotoru jsou vstupními parametry požadovaný točivý moment a úhlová rychlost na hřídeli. Výstupní hodnotou je pak výkon elektromotoru. Je-li hodnota výkonu větší než nula, elektromotor pracuje v motorickém režimu, zatímco když je hodnota výkonu záporná, tak pracuje elektromotor v generátorickém režimu. Pro výpočet výkonu elektromotoru pro jednotlivé pracovní režimy platí následující rovnice:

$$P_{EM}(t) = \frac{M_{EM}(t) \cdot \omega_{EM}(t)}{\eta_{EM}(\omega_{EM}(t), M_{EM}(t))}, \quad M_{EM}(t) \geq 0 \quad (30)$$

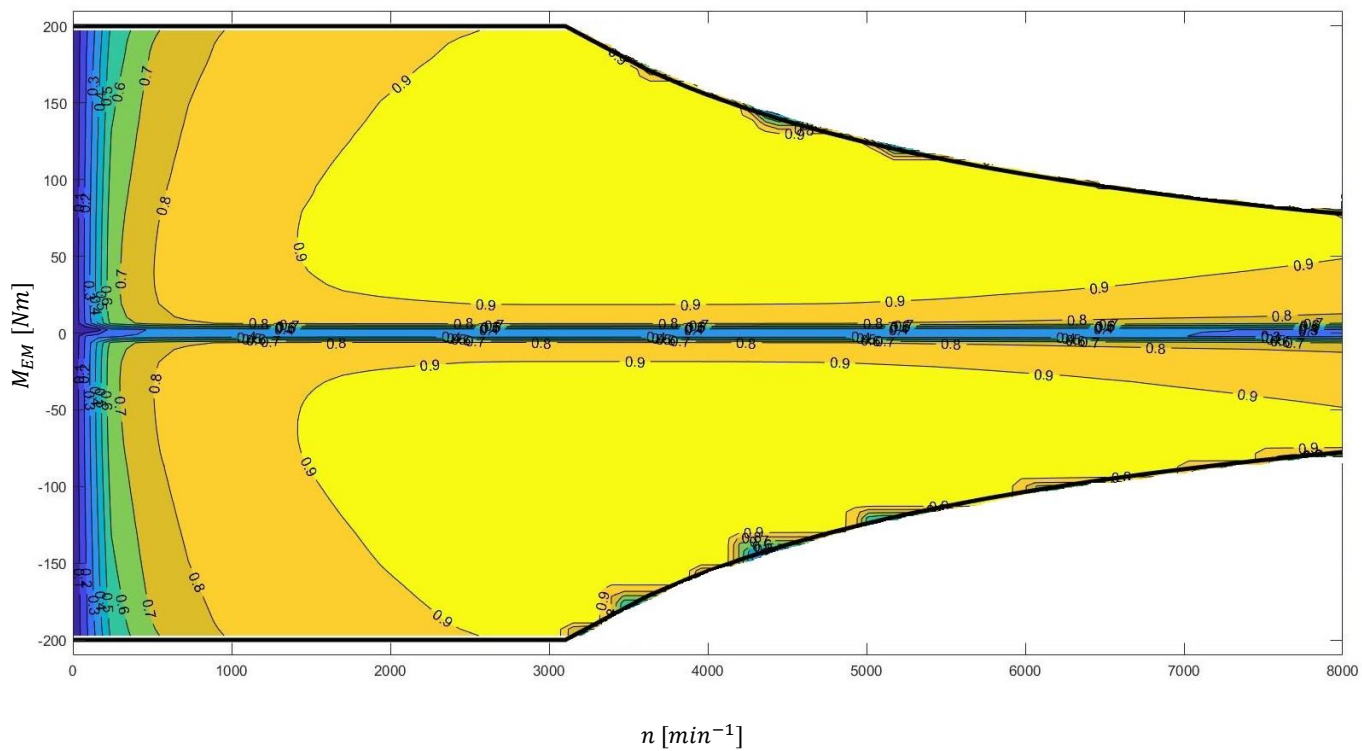
$$P_{EM}(t) = M_{EM}(t) \cdot \omega_{EM}(t) \cdot \eta_{EM}(\omega_{EM}(t), M_{EM}(t)), \quad M_{EM}(t) \leq 0 \quad (31)$$

Kde vztah (30) platí pro motorický režim a rovnice (31) pro generátorický režim. Účinnost elektromotoru η_{EM} je závislá na aktuální úhlové rychlosti a točivém momentu. Hodnota účinnosti se získává z mapy účinnosti elektromotoru, která v algoritmu slouží jako look-up table. Ukázka mapy účinnosti elektromotoru, která je používaná v algoritmu řízení je vidět na obrázku č. 13.

Pro vstupní hodnoty točivého momentu a úhlové rychlosti ještě platí následující limity:

$$M_{EM,min}(\omega_{EM}(t)) \leq M_{EM}(t) \leq M_{EM,max}(\omega_{EM}(t)) \quad (32)$$

$$0 \leq \omega_{EM}(t) \leq \omega_{EM,max}(t) \quad (33)$$



Obrázek 13: Mapa účinnosti elektromotoru

Na obrázku č. 13 je znázorněná mapa účinnosti elektromotoru používaného v rámci této práce. Zároveň je zde vyznačený i maximální točivý moment elektromotoru v závislosti na otáčkách.

V tabulce č. 2 jsou uvedeny základní parametry použitého elektromotoru.

Výkon elektromotoru	65 kW
Maximální točivý moment	200 Nm
Maximální otáčky motoru	8000 min ⁻¹
Jmenovité otáčky	3104 min ⁻¹

Tabulka 2: Základní parametry elektromotoru



3.1.6 Spalovací motor

Poslední důležitou součástí pohonného systému, kterou je potřeba zahrnout do modelu vozidla, je spalovací motor. Model bere v úvahu úplnou charakteristiku spalovacího motoru, která je dána točivým momentem a úhlovou rychlostí. Pro tyto dva parametry opět platí limity, jako tomu bylo u elektromotoru.

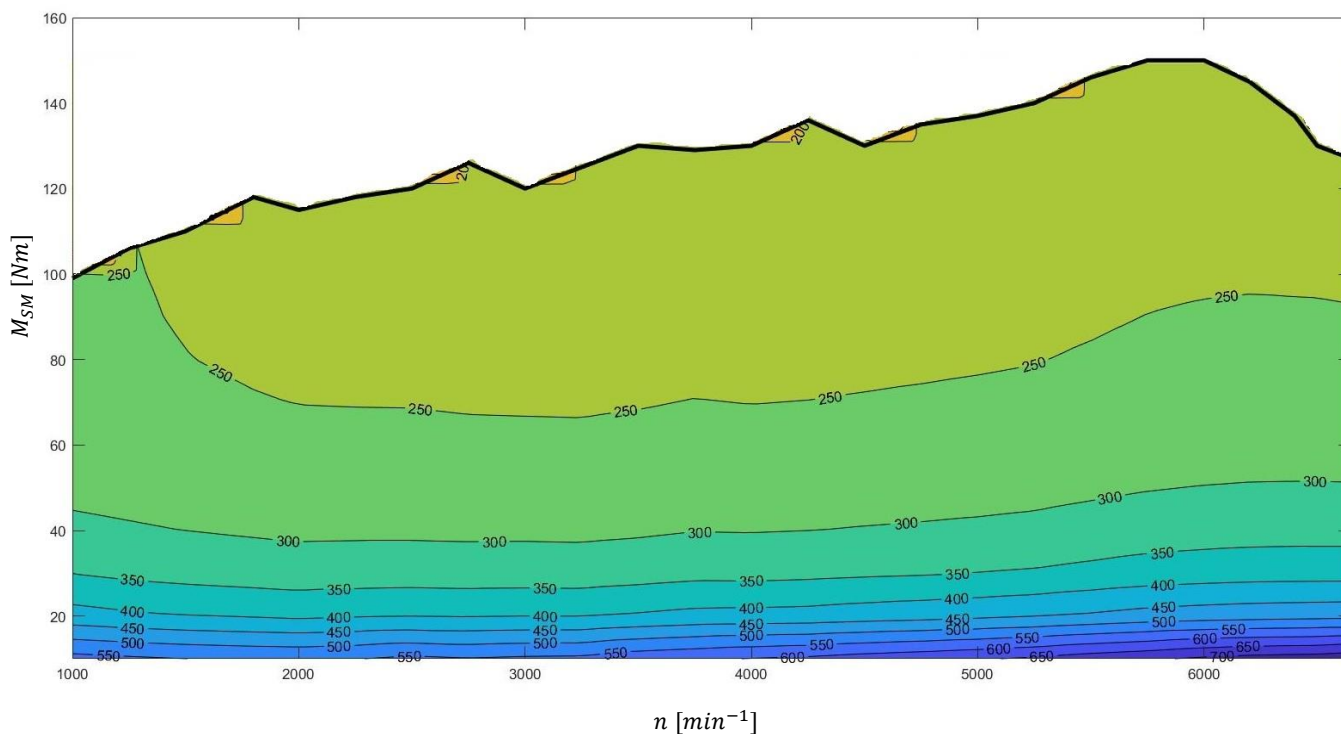
$$0 \leq M_{SM}(t) \leq M_{SM,max}(\omega_{SM}(t)) \quad (34)$$

$$\omega_{SM,max}(t) \leq \omega_{SM}(t) \leq \omega_{SM,max}(t) \quad (35)$$

Pro samotný řídicí algoritmus dynamického programování je nutné znát aktuální spotřebu paliva v každém časovém kroku. Pro výpočet spotřeby paliva platí následující rovnice:

$$\dot{m}_p(t) = m_p(\omega_{SM}(t), M_{SM}(t)) \cdot M_{SM}(t) \cdot \omega_{SM}(t) \quad (36)$$

Kde m_p je měrná spotřeba paliva udávaná v g/kWh . Hodnota měrné spotřeby paliva se získává na základě úhlové rychlosti a točivého momentu z úplné charakteristiky motoru, která slouží jako look-up table. Ukázka úplné charakteristiky spalovacího motoru, která byla použita v rámci této práce, je zobrazena na obrázku č. 14.



Obrázek 14: Úplná charakteristika spalovacího motoru

Na obrázku č. 14 je také vyznačena křivka maximálního točivého momentu v závislosti na otáčkách.



V tabulce č. 3 jsou uvedeny základní parametry použitého spalovacího motoru.

Výkon spalovacího motoru	96 kW
Maximální točivý moment	150 Nm
Maximální otáčky motoru	6000 min ⁻¹
Volnoběžné otáčky motoru	1000 min ⁻¹
Jmenovité otáčky motoru	4000 min ⁻¹

Tabulka 3: Základní parametry spalovacího motoru

3.2 Strategie řazení

Pro napodobení reálné jízdy je nutné k modelu vozidla implementovat ještě strategii řazení, která na základě zvolených kritérií volí převodový stupeň. Co se týče minimalizace spotřeby paliva, tak výběr optimálního převodového poměru by také mohl být součástí optimalizačního algoritmu, protože ovlivňuje účinnost spalovacího motoru a elektromotoru. V rámci této práce se ale použila samostatná strategie řazení, především z důvodu zjednodušení a razantního zvýšení výpočetní náročnosti, které by implementace volby optimálního převodového stupně do řídicího algoritmu přinesla.

Použitá řadící strategie je založená na řadící strategii WLTP (Worldwide Harmonized Light-Duty Vehicles Test Procedure). Ta na základě aktuálních výkonových požadavků zařadí nejvyšší možný převodový stupeň při splnění dvou základních kritérií. Co se týče řazení chová se auto jako klasické konvenční vozidlo bez elektromotoru. [21] [23]

Prvním kritériem je, že dostupný výkon motoru v čase t pro převodový stupeň i musí být větší než požadovaný výkon v čase t .

$$P_{av}(t, i) \geq P_{req}(t) \quad (37)$$

Kde P_{av} je dostupný výkon pro daný převodový stupeň a P_{req} je požadovaný výkon, jehož hodnotu získáme na základě jízdních odporů. Dostupný výkon motoru dostaneme pomocí následujícího výpočtu:

$$P_{av}(t, i) = P_{wot}(n(t, i)) \cdot SM \quad (38)$$

Kde P_{wot} je dostupný výkon při otáčkách $n(t, i)$ při plném zatížení. Jeho hodnotu získáme na základě otáček interpolací z úplné charakteristiky motoru, viz. obrázek č. 14. Ve vztahu (38) figuruje ještě SM , což je bezpečnostní rezerva, která se ve většině případů bere jako 10%.



Druhým kritériem je, že otáčky motoru musí být uvnitř stanovených limit. Pro první dva převodové stupně jsou to pouze limity minima a pro vyšší stupně už jsou limity minima i maxima. Pro první a druhý převodový stupeň platí následující rovnice:

$$n_{min} = \max[1,15 \cdot n_{idle}, 0,03 \cdot (n_{rated} - n_{idle}) + n_{idle}] \quad (39)$$

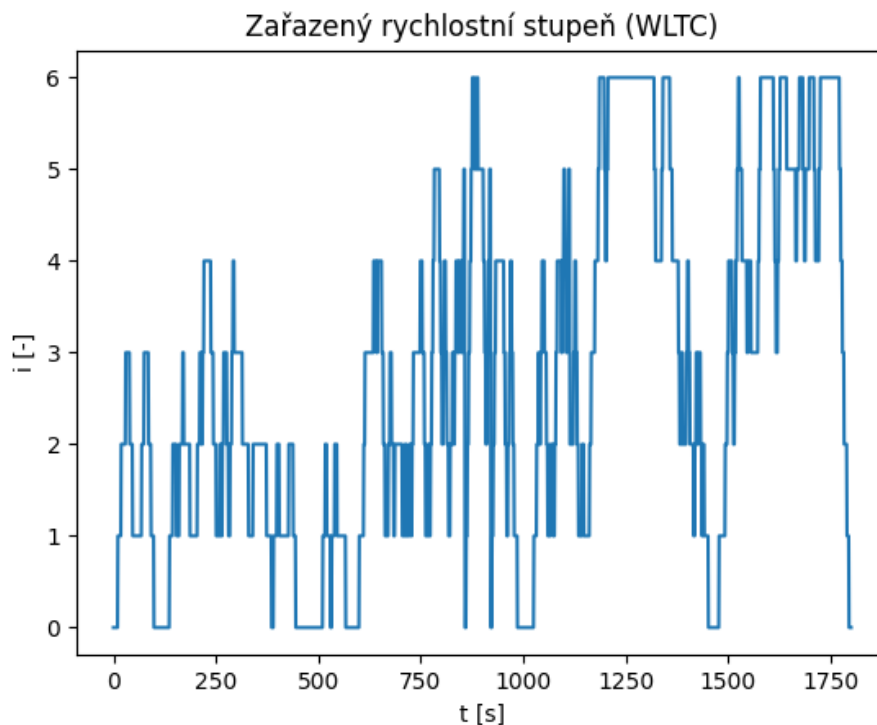
Kde n_{idle} jsou volnoběžné otáčky spalovacího motoru a n_{rated} jsou jmenovité otáčky motoru.

Pro vyšší převodové stupně platí následující limity:

$$n_{min} = 0,125 \cdot (n_{rated} - n_{idle}) + n_{idle} \quad (40)$$

$$n_{max} = 1,2 \cdot (n_{rated} - n_{idle}) + n_{idle} \quad (41)$$

Podle strategie WLTP je zakázáno přeskokování převodových stupňů a převodový stupeň by měl být zařazený alespoň 3 vteřiny. Tyto pravidla jsou také součástí strategie řazení použité v rámci této práce. Na grafu č. 1 je na ukázkou vidět, jak vypadá průběh řazení podle výpočtu pro jízdní cyklus WLTC. [21] [23]



Graf 1: Průběh řazení podle strategie WLTP



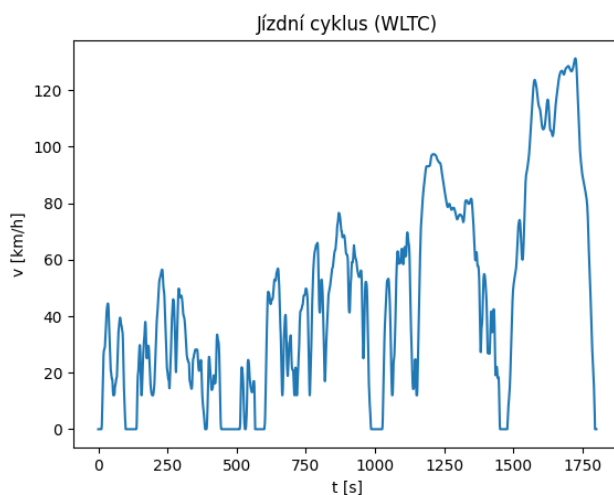
3.3 Jízdní cykly

Jízdní cykly představují hlavní zdroj vstupních hodnot do modelu hybridního vozidla. Na základě dat získaných z jízdního cyklu se počítá například požadovaný točivý moment, který je jednou z nejdůležitějších veličin pro algoritmus řízení hybridního pohonu. V rámci této práce byly použity celkem čtyři jízdní cykly. Jedná se o běžně dostupné testovací a homologační cykly, které simulují různé druhy jízdy – jízdu po městě i mimo město, nebo jízdu po rychlostní komunikaci. Použité jízdní cykly jsou uvedeny v následujícím seznamu:

- WLTC
- NEDC
- FTP 75
- FTP HW

WLTC

The Worldwide harmonized Light vehicles Test Cycles, zkratkou WLTC, je prvním jízdním cyklem použitým v této práci. Jízdní cyklus je součástí WLTP (World harmonized Light-duty vehicles Test Procedure), což je celosvětová harmonizovaná norma pro určování úrovní znečišťujících látek, emisí CO₂ a spotřeby paliva hybridních, ale i čistě elektrických vozidel. Testovací procedurou je tedy cyklus WLTC, který se na základě měrného výkonu vozidla dělí do tří tříd – 1 až 3. Pro účely této práce se použil cyklus třídy 3. Samotný jízdní cyklus je pak rozdělen na čtyři fáze, kde každá z nich simuluje odlišné jízdní podmínky – městské, příměstské, venkovské a dálniční. Rychlostní profil cyklu WLTC je vidět na grafu č. 2. [21]

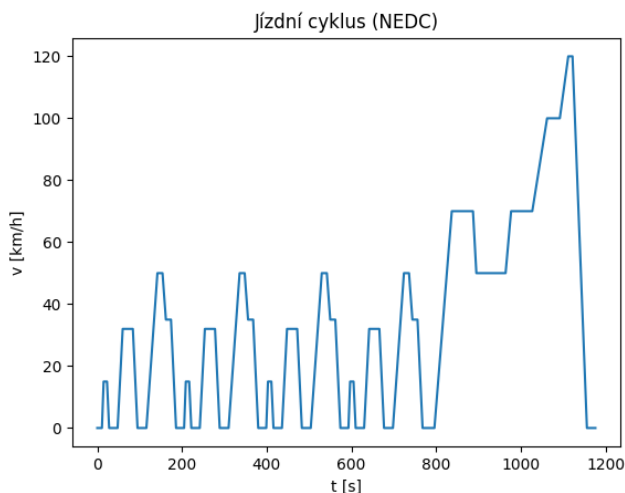


Graf 2: Rychlostní profil cyklu WLTC



NEDC

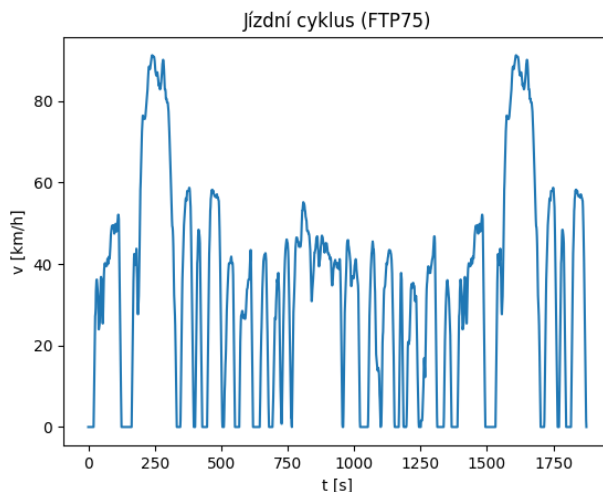
New European Driving Cycle (NEDC) je dalším jízdním cyklem používaným v rámci této práce. Jedná se opět o testovací proceduru, navrženou pro hodnocení emisí a spotřeby paliva osobních automobilů. Cyklus byl primárně vytvořený pro konvenční vozidla, ale je používán i pro hybridy. Rychlostní profil cyklu NEDC je vidět na grafu č. 3. [22]



Graf 3: Rychlostní profil cyklu NEDC

FTP 75

Dalším používaným jízdním cyklem je FTP 75. Ten je součástí EPA Federal Test Procedure, což je série testovacích cyklů, které se využívají ve Spojených Státech Amerických pro měření výfukových emisí a spotřeby paliva osobních automobilů. Pomocí tohoto cyklu se simuluje především městská a příměstská jízda vozidla. Rychlostní profil cyklu FTP 75 je vidět na grafu č. 4.

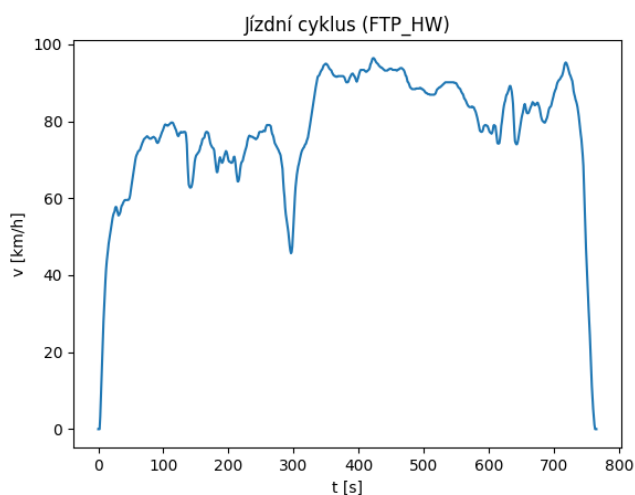


Graf 4: Rychlostní profil cyklu FTP 75



FTP HW

Dalším jízdním cyklem, který je součástí EPA Federal Test Procedure, je FTP HW. Tento jízdní cyklus simuluje jízdu po rychlostní komunikaci bez zastavení vozidla. Rychlostní profil cyklu FTP HW je zobrazen na grafu č. 5.



Graf 5: Rychlostní profil cyklu FTP HW

V následující tabulce je vidět porovnání vlastností jednotlivých jízdních cyklů používaných v rámci této práce.

Jízdní cyklus	WLTC	NEDC	FTP 75	FTP HW
<i>Celkový čas jízdy [s]</i>	1800	1177	1875	766
<i>Délka jízdního cyklu [km]</i>	23,26	10,76	17,77	16,51
<i>Maximální rychlost [km/h]</i>	131,3	120	91,2	96,4
<i>Průměrná rychlost [km/h]</i>	46,5	32,9	34,1	77,6
<i>Maximální zrychlení [m/s^2]</i>	1,75	1,04	1,47	1,43
<i>Podíl stání vozidla [%]</i>	13	25	19,1	0,8

Tabulka 4: Porovnání jízdních cyklů



3.4 Algoritmus dynamického programování

Princip strategie řízení s využitím dynamického programování byl již naznačen v kapitole 2.2.1, ale v této části bude algoritmus vysvětlen více do hloubky a bude naznačeno přímé použití v řízení pohonného systému hybridního vozidla. Jen pro zopakování, jedná se o offline strategii řízení, takže musí být informace o jízdním cyklu dopředu známy, a algoritmus zaručuje globálně optimální řešení, což znamená, že cílem optimalizace není minimalizovat výkonnostní měřítko v každém časovém okamžiku, ale na určitém časovém intervalu, nebo na celém jízdním cyklu. Globálně optimální řešení je tedy takové řešení, které minimalizuje výkonnostní měřítko na celé trase, tedy globálně.

3.4.1 Odvození algoritmu

Optimální řízení, nebo ovládání je definováno pomocí vztahu (42), kde f je kontrolní funkce a x je stavová veličina.

$$u^*(t) = f(x(t), t) \quad (42)$$

Optimalizovaný systém můžeme popsat pomocí diferenciální rovnice 1. stupně, která má následující formu:

$$\frac{d}{dt}[x(t)] = a \cdot x(t) + b \cdot u(t) \quad (43)$$

Poté je dáno výkonnostní měřítko J , které bývá častěji označováno jako cenová funkce, a ta se má během optimalizace minimalizovat.

$$J = x^2(t_f) + \lambda \int_{t_0}^{t_f} u^2(t) dt \quad (44)$$

Kde t_f je konečný čas a λ je váhový faktor, který kompenzuje důležitost jednotlivých členů. Stavové a řídicí hodnoty jsou zde s druhou mocninou, protože kladné a záporné hodnoty těchto veličin jsou stejně důležité. [10]

Před samotným numerickým postupem dynamického programování je nutné udělat ještě několik úprav. Nejdříve je nutné aproximovat diferenciální rovnici systému pomocí diferenční rovnice a integrál v cenové funkci se musí nahradit sumací. Toho jde dosáhnout rozdělením časového intervalu $t_0 \leq t \leq t_f$ na N kroků jako Δt . Potom můžeme nahradit vztah (43) rovnicí (45) a po úpravách získáme vztah (46):

$$\frac{x(t + \Delta t) - x(t)}{\Delta t} \approx a \cdot x(t) + b \cdot u(t) \quad (45)$$

$$x(k + 1) = [1 + a \cdot \Delta t] \cdot x(k) + b \cdot \Delta t \cdot u(k), \text{ kde } k = 0, 1, \dots, N - 1 \quad (46)$$



Podobným způsobem nahradíme i cenovou funkci a dostaneme následující vztah:

$$J = x^2(N) + \lambda \cdot \Delta t \cdot \sum_{k=0}^{N-1} u^2(k) \quad (47)$$

Pro představení výpočetního procesu uvažujme poslední dvě fáze procesu ($N = 2$). Prvním krokem ve výpočetním procesu je najít optimální řízení pro poslední fázi procesu. Toho se dosáhne vyzkoušením všech přípustných řídicích hodnot v každé přípustné stavové hodnotě. Pro zajištění uskutečnitelnosti výpočetního procesu je nutné stavové a řídicí hodnoty diskretizovat. Optimální řízení pro každou stavovou hodnotu je pak to, které dává minimální cenu operace. Postup pro určení optimálního řízení v poslední fázi se pak skládá z těchto dvou bodů:

1. Pro $k = 1$, vezme se jedna stavová hodnota $x(1)$ a vyzkoušejí se v ní všechny řídicí hodnoty $u(1)$. Optimální řízení pro tuto stavovou hodnotu je to, které dává minimální cenu operace.
2. Proces z bodu 1. se zopakuje pro všechny zbylé stavové hodnoty $x(1)$.

Pro předposlední fázi platí $k = 0$. V každé stavové hodnotě $x(0)$ se opět vyzkouší všechny řídicí hodnoty $u(0)$ a vypočítá se trajektorie z $x(0)$ do $x(1)$ a cena operace J_{01} pro každou iteraci. Jelikož už známe hodnotu $x(1)$ na konci každé této trajektorie, můžeme pak následovat optimální trajektorii v poslední fázi díky hodnotám vypočítaným v předchozí fázi. [10]

To je možné matematicky znázornit pomocí následující rovnice:

$$C_{02}^*(x(0), u(0)) = J_{01}(x(0), u(0)) + J_{12}^*(x(1)) \quad (48)$$

A cenová funkce optimální trajektorie je pak dána tímto vztahem:

$$J_{02}^*(x(0)) = \min_{u(0)} [C_{02}^*(x(0), u(0))] \quad (49)$$

Kde

$C_{02}^*(x(0), u(0))$ je minimální cena operace za poslední dvě fáze pro jednu diskretizovanou hodnotu $x(0)$ po vyzkoušení hodnot $u(0)$.

$J_{01}(x(0), u(0))$ je cena operace v intervalu od $k = 0$ do $k = 1$ pro specifické hodnoty $x(0)$ a $u(0)$.

$J_{12}^*(x(1))$ je cena optimální trajektorie poslední fáze, která je funkcí stavu $x(1)$.

$J_{02}^*(x(0))$ je minimální cena trajektorie přes poslední dvě fáze pro specifickou hodnotu stavu $x(0)$. [10]



Jelikož má optimalizační úloha zpravidla více než dvě fáze, je nutné tento postup opakovat pro všechny fáze, na které je časový interval rozdělený. Obecně pro stanovení optimálního řízení v čase $t = k \cdot \Delta t$ v N-fázovém procesu dostanou vztahy (48) a (49) následující formu:

$$C_{kN}^*(x(k), u(k)) = J_{k,k+1}(x(k), u(k)) + J_{k+1,N}^*(x(k+1)) \quad (50)$$

$$J_{kN}^*(x(k)) = \min_{u(k)} [C_{kN}^*(x(k), u(k))] \quad (51)$$

Spojením těchto dvou rovnic a následnými úpravami získáme tzv. funkční rovnici dynamického programování:

$$J_{N-K,N}^*(x(N-K)) = \min_{u(N-K)} \{g_D(x(N-K), u(N-K)) + J_{N-(K-1),N}^*(a_D(x(N-K), u(N-K)))\} \quad (52)$$

Rovnice znázorňuje rekurentní vztah, který chceme řešit. K představuje fázi procesu. Jelikož známe $J_{N-(K-1),N}^*$, což je optimální cenová funkce pro $(K-1)$ fázi, můžeme pak generovat $J_{N-K,N}^*$, což je optimální cena pro K -tou fázi. Pro začátek celého procesu se začne fází $K=0$ a vygeneruje se $J_{N,N}^* \triangleq J_{N,N}$, kde $J_{N,N}$ je zvolená koncová hodnota. Poté je možné získat optimální cenu první fáze pomocí hodnoty $J_{N,N}^*$ a to se pak opakuje pro všechny fáze. [10]



3.4.2 Výpočetní proces

V této kapitole bude podrobně vysvětlen samotný princip výpočetního procesu algoritmu dynamického programování, který vychází z [10]. Nejdříve je nutné zavést předpoklady a provést přípravu na výpočet.

Předpokládáme, že systém je popsáný diferenční rovnicí v následujícím tvaru:

$$x(k-1) = a_D(x(k), u(k)), \quad \text{kde } k = 0, 1, \dots, N-1 \quad (53)$$

Cílem optimalizace je minimalizovat výkonnostní měřítko J , které je dáno následující rovnicí:

$$J = h(x(N)) + \sum_{k=0}^{N-1} g_D(x(k), u(k)) \quad (54)$$

Aplikace dynamického programování na tuto optimalizační úlohu vede k rekurzivní rovnici:

$$J_{N-K,N}^*(x(N-K)) = \min_{u(N-K)} \{g_D(x(N-K), u(N-K)) + J_{N-(K-1),N}^*(a_D(x(N-K), u(N-K)))\} \quad (55)$$

Kde výchozí hodnotou je $J_{N,N}^*(x(N)) = h(x(N))$.

Začne se načtením dat, což jsou v podstatě 3 for-cykly. Jde o množství stavů neboli časových kroků, čemuž odpovídá K , kde $K = 1, 2, \dots, N$. Dále je to množství stavových hodnot, kde je třeba použít diskretizace. Tomu odpovídá index i , kde $i = 1, 2, \dots, S$. V posledním cyklu jsou řídicí hodnoty, u kterých je také nutná diskretizace. Těm odpovídá index j , kde $j = 1, 2, \dots, C$. Připraví se matice $S \times N$ *Cost* a *Uopt* a u matice *Cost* se nejdříve nastaví hodnoty proměnné *COSMIN* (jako minimální cena) na hodně vysoké číslo, se kterým se následně bude porovnávat vypočítaná hodnota. Pro ukládání řídicích hodnot se zavede proměnná *UMIN*.

Prvním krokem výpočetního procesu je zavedení, nebo výpočet hodnot $J_{N,N}^*(x^i(N))$, které budou použity pro začátek výpočtu rekurzivní rovnice.

V dalším kroku se nastaví $K = 1$ a zvolí se první zkušební stav pro $i = 1$. Následně se každá řídicí hodnota $u^j(N-K)$, pro $j = 1, 2, \dots, C$, vyzkouší v této stavové hodnotě $x^i(N-K)$, pro určení následující stavové hodnoty $x^{i,j}(N-K+1)$. Ta se použije pro vyhledání odpovídající hodnoty $J_{N-(K-1),N}^*(x^{i,j}(N-K+1))$ v paměti počítače. K tomu je většinou potřeba interpolace. [10]



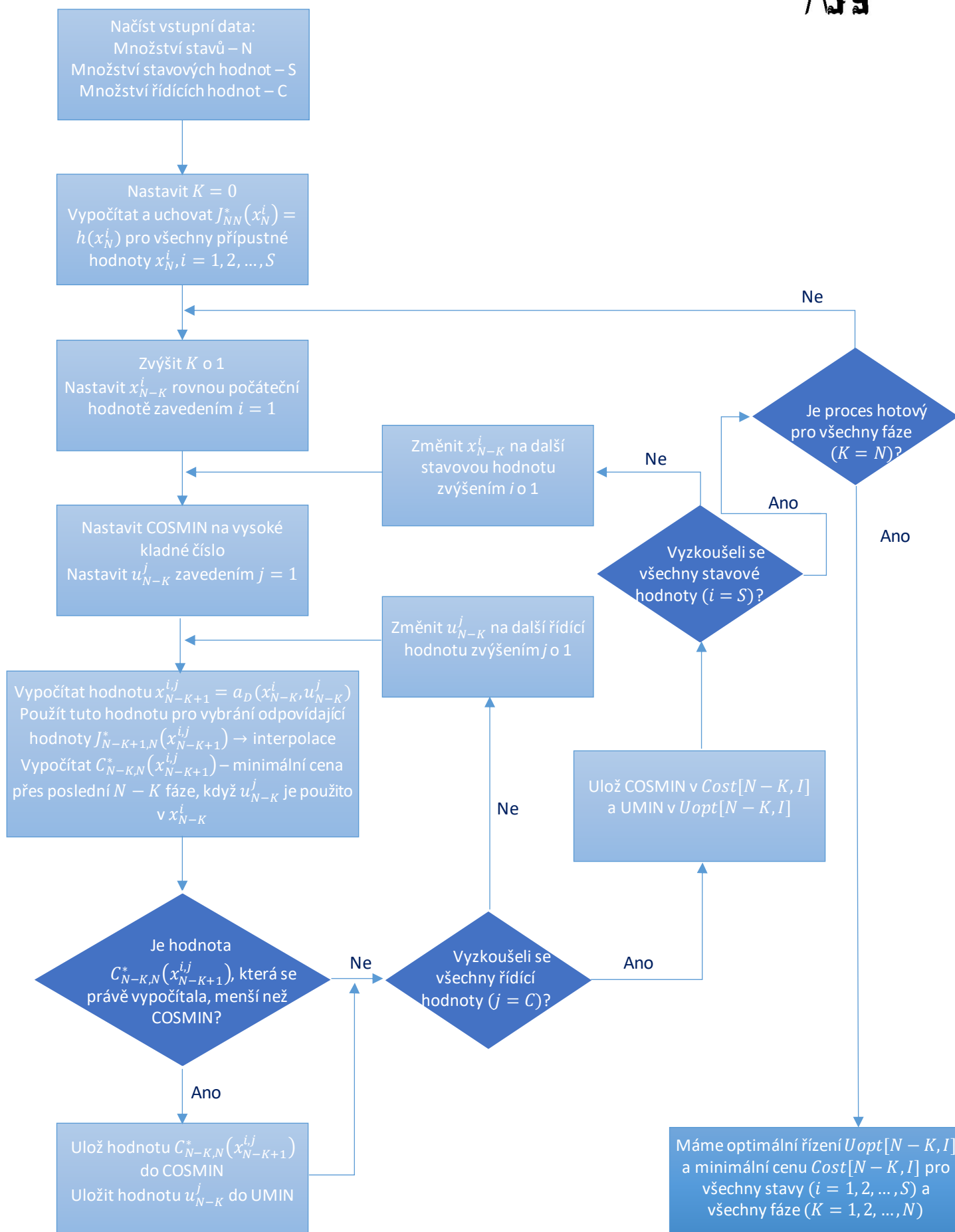
Hodnotu $J_{N-(K-1),N}^*(x^{i,j}(N-K+1))$ použijeme do následující rovnice:

$$C_{N-K,N}^*(x^i(N-K), u^j(N-K)) = J_{N-K,N}(x^i(N-K), u^j(N-K)) + J_{N-(K-1),N}^*(x^{i,j}(N-K+1)) \quad (56)$$

Což je minimální cena operace přes posledních K fází z N fázového procesu, za předpokladu že řídicí hodnota $u^j(N-K)$ je použita ve stavové hodnotě $x^i(N-K)$. Cílem je najít hodnotu $u^j(N-K)$ takovou, která nám dá hodnotu $J_{N-K,N}^*(x^i(N-K))$, což je minimum z $C_{N-K,N}^*(x^i(N-K), u^j(N-K))$. Pouze nejnižší hodnota $C_{N-K,N}^*(x^i(N-K), u^j(N-K))$ a její přidružená řídicí hodnota se uchovávají v paměti. Proto když je řídicí hodnota použita ve stavu $x^i(N-K)$, $C_{N-K,N}^*(x^i(N-K), u^j(N-K))$ která tak vznikne se porovnává s proměnnou $COSMIN$. Když je $C_{N-K,N}^*(x^i(N-K), u^j(N-K)) < COSMIN$, tak se současná hodnota proměnné $COSMIN$ vymění za novou menší hodnotu. Řídicí hodnota, která odpovídá hodnotě $COSMIN$ se zachovává jako proměnná $UMIN$. [10]

Když jsou všechny řídicí hodnoty vyzkoušeny pro stav $x^i(N-K)$, tak hodnoty uložené v proměnných $COSMIN$ a $UMIN$ jsou přetransformovány do matic $Cost(N-K, I)$ a $Uopt(N-K, I)$. Tato procedura se opakuje pro každou diskretizovanou stavovou hodnotu. Poté se zvýší K o jedna a proces se opakuje až do $K = N$. [10]

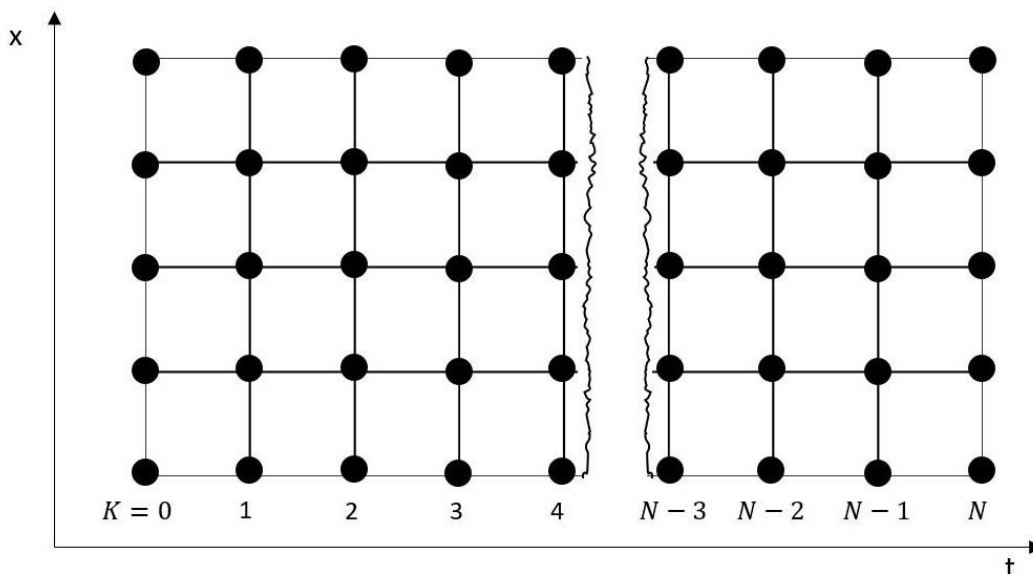
Tento proces představuje zpětnou rekurzi a na obrázku č. 15 je vidět vývojový diagram výpočetního procesu.



Obrázek 15: Vývojový diagram výpočetního procesu algoritmu

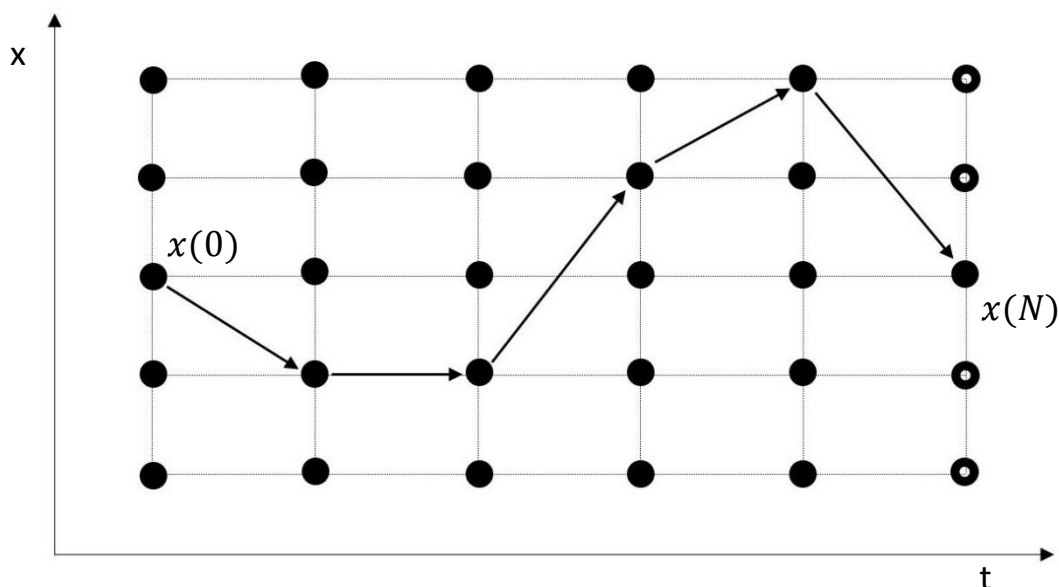


Výsledkem výpočetního procesu zpětné rekurze je hodnota pro optimální řízení u^* a minimální cenová funkce v každém bodě na $(n - 1)$ -dimenzionální stavově-časové mřížce. Ta slouží jako look-up table pro následující výpočet. Stavově-časová mřížka je znázorněná na obrázku č. 16.



Obrázek 16: Stavově-časová mřížka

Po dokončení zpětné rekurze může nastat výpočet dopředné rekurze, tedy od času $t = 0$. Dopředná rekurze probíhá tak, že pro výpočet optimální řídicí sekvence pro danou počáteční podmínku se z paměti vezme, na základě tohoto počátečního stavu, řídicí hodnota $u^*(0)$ a její odpovídající cena. Poté výpočtem stavové rovnice určíme stav systému ve fázi $K = 1$, na základě použití řídicí hodnoty $u^*(0)$ ve fázi $K = 0$. Získaná stavová hodnota $x(1)$ se potom použije jako vstupní hodnota do look-up table, čímž získáme řídicí hodnotu $u^*(1)$. A tímto způsobem proces pokračuje až do fáze $K = N$. [10]



Obrázek 17: Dopředná rekurze



3.4.3 Implementace algoritmu na řízení hybridního pohonu

V první řadě je nutné definovat optimalizační úlohu, kterou bude algoritmus dynamického programování řešit. Jelikož se jedná o optimalizaci řízení hybridního pohonného systému, bude se jednat o optimalizaci spotřeby paliva. Cílem tedy je, aby se pomocí algoritmu našla řídicí hodnota u^* v každém časovém intervalu, která bude minimalizovat výkonnostní měřítko, které představuje spotřebu paliva za celý jízdní cyklus. To je matematicky popsáno následujícím vztahem:

$$J_{(u)} = \int_{t_0}^{t_f} \dot{m}_p(u(t), t) dt \quad (57)$$

U stavových hodnot je jedinou kontrolovatelnou veličinou, která je součástí optimalizace, stav nabití baterie, protože rychlost vozidla je dána rychlostním profilem jízdního cyklu. Stavovou hodnotou je tedy pouze stav nabití baterie.

$$x(t) = SOC(t) \quad (58)$$

Stav nabití baterie je pro výpočet nutné diskretizovat pro zaručení uskutečnitelnosti výpočetního procesu. Velikost jednoho kroku je označena jako $dSOC$.

Jak už bylo řečeno v kapitole 3.2, tak řazení rychlostních stupňů je řešeno samostatnou strategií založenou na rychlosti vozidla a požadovaném točivém momentu a není tedy součástí řídicího algoritmu. To tedy znamená, že jedinou řídicí hodnotou v algoritmu je točivý moment elektromotoru.

$$u(t) = M_{EM}(t) \quad (59)$$

Přípustný točivý moment elektromotoru je pro výpočet nutné diskretizovat. Velikost jednoho kroku se označuje jako dM_{EM} .

Jelikož je řídicí hodnotou v algoritmu točivý moment elektromotoru, je pak možné na základě požadovaného točivého momentu pomocí vztahu (60) určit, jakou hodnotu točivého momentu musí dodat spalovací motor.

$$M_{SM}(t) = M_{req}(t) - u(t) \quad (60)$$



Pomocí těchto vztahů dostáváme výslednou rovnici pro výpočet výkonnostního měřítka, které má v obecném tvaru následující formu:

$$J_{(u)} = \int_{t_0}^{t_f} m_p(\omega(t), M_{SM}(u(t), t)) \cdot M_{SM}(u(t), t) \cdot \omega(t) dt \quad (61)$$

Kde m_p je opět měrná spotřeba paliva, která je závislá na otáčkách a točivém momentu spalovacího motoru a získává se z úplné charakteristiky spalovacího motoru, viz. obrázek č. 14.

Pro naznačení výpočtu s parametry definovanými v této kapitole použijeme upravenou formu rekurzivní rovnice, která byla definována v předchozí kapitole, a je popsána následujícím vztahem:

$$C_{N-K,N}^*(x^i(N-K), u^j(N-K)) = J_{N-K,N}(x^i(N-K), u^j(N-K)) + J_{N-(K-1),N}^*(x^{i,j}(N-K+1)) \quad (62)$$

Kde indexy N a K vyjadřují čas – N je celkový počet časových intervalů a K představuje aktuální fázi. Indexu i odpovídá diskretizace stavu nabití baterie a indexu j odpovídá diskretizace přípustného točivého momentu elektromotoru.

Pokud bychom si rozebrali jednotlivé členy rovnice, tak můžeme začít u členu $J_{N-K,N}(x^i(N-K), u^j(N-K))$. Jedná se o cenovou funkci v aktuální fázi $N-K$, která je závislá na stavových a řídicích hodnotách. Hodnota cenové funkce se vypočítá na základě vztahu (61) následující rovnicí:

$$\begin{aligned} J_{N-K,N}(x^i(N-K), u^j(N-K)) \\ = m_p(\omega(N-K), M_{req} - u^j(N-K)) \cdot (M_{req} - u^j(N-K)) \cdot \omega(N-K) \end{aligned} \quad (63)$$

Druhým členem rovnice je $J_{N-(K-1),N}^*(x^{i,j}(N-K+1))$ a jedná se o optimální cenu v příští fázi. Pro získání její hodnoty je nutné určit stavovou hodnotu $x^{i,j}(N-K+1)$. Tu je možné vypočítat pomocí následujících vztahů:

$$x^{i,j}(N-K+1) = SOC_{x^i} + \Delta SOC \quad (64)$$

Kde SOC_{x^i} je stav nabití baterie při daném stavu x^i a ΔSOC je změna stavu nabití baterie za jeden časový krok a vypočítá se podle následujících vztahů (65) až (67).



$$\Delta SOC = 0, \quad \text{když } |u^j(N-K)| < dM_{EM} \quad (65)$$

$$\Delta SOC = - \frac{U_{OC} - \sqrt{U_{OC}^2 - 4 \cdot R_{Bat} \cdot \frac{u^j(N-K) \cdot \omega(N-K)}{\eta_{EM}(\omega(N-K), u^j(N-K))}}}{2 \cdot R_{Bat} \cdot C_{Bat}}, \text{ když } u^j(N-K) > 0 \quad (66)$$

$$\Delta SOC = - \frac{U_{OC} - \sqrt{U_{OC}^2 - 4 \cdot R_{Bat} \cdot u^j(N-K) \cdot \omega(N-K) \cdot \eta_{EM}(\omega(N-K), u^j(N-K))}}{2 \cdot R_{Bat} \cdot C_{Bat}}, \quad (67)$$

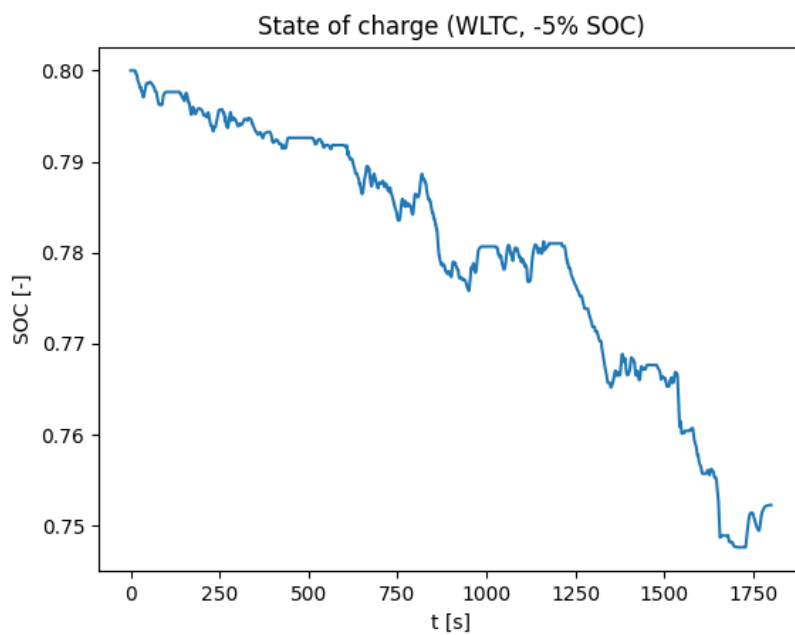
když $u^j(N-K) < 0$

Vztah (65) platí, když absolutní hodnota $u^j(N-K)$ je menší než diskretizace točivého momentu elektromotoru, protože ΔSOC je zanedbatelná. Druhý vztah platí, když elektromotor pracuje jako motor (ΔSOC má zápornou hodnotu) a vztah (67) platí, když elektromotor pracuje v generátorickém režimu (ΔSOC má kladnou hodnotu).

Hodnota optimální cenové funkce $J_{N-(K-1),N}^*(x^{i,j}(N-K+1))$ se pak na základě stavové hodnoty $x^{i,j}(N-K+1)$ určí pomocí interpolace z hodnot uložených v paměti z předchozích výpočtů.

Součtem těchto dvou členů získáme $C_{N-K,N}^*(x^i(N-K), u^j(N-K))$, což je minimální cena operace přes posledních K fází z N fázového procesu, za předpokladu že řídicí hodnota $u^j(N-K)$ je použita ve stavové hodnotě $x^i(N-K)$. Ta se porovnává s hodnotou proměnné $COSMIN$, jak už bylo zmíněno v předchozí kapitole.

Počáteční podmínkou pro výpočet je zvolená hodnota stavu nabití baterie po ujetí jízdního cyklu. Výstupem ze simulace, který je při řízení hybridního pohonného systému stěžejní, je především průběh stavu nabití baterie a hodnota spotřeby paliva. V průběhu simulace se ale samozřejmě počítají i další parametry. Důkladný přehled výsledků simulace pro jednotlivé jízdní cykly a různé počáteční podmínky bude znázorněn v kapitole 5. Na grafu č. 7 je alespoň na ukázkou vidět průběh stavu nabití baterie po simulaci na jízdním cyklu WLTC, kde stav nabití baterie na začátku jízdního cyklu byl 80% a zvolená hodnota SOC po ujetí trasy byla nastavena na 75%.



Graf 6: Průběhu SOC po simulaci na jízdním cyklu WLTC



4 Metody zlepšení výpočetního času základního algoritmu

Další fází této práce je optimalizace základního algoritmu dynamického programování z hlediska výpočetního času. Jak již bylo řečeno, tak základním problémem řízení hybridního pohonného systému pomocí algoritmu s implementací dynamického programování je vysoká výpočetní náročnost. Řešení komplexního optimalizačního problému pomocí dynamického programování může trvat řádově i hodiny, proto bylo jedním z cílů této práce vytvořit algoritmus, který dokáže provést výpočet optimálního řízení s dostatečnou přesností za výpočetní čas, který je v rámci desítek sekund, nebo maximálně několika minut. Z tohoto důvodu se při optimalizaci bude jednat především o možnosti snížení výpočetního času algoritmu se zachováním dostatečné přesnosti.

Výpočetní rychlost algoritmu přirozeně závisí na počtu časových intervalů a velikosti diskretizace stavu nabití baterie a přípustného točitého momentu elektromotoru. Tyto faktory ale nemůžeme považovat za možnost zrychlení výpočtu. Pro všechna porovnání byly hodnoty diskretizace nastaveny na tyto hodnoty - $dM_{EM} = 5 Nm$ a $dSOC = 0,25\%$. Existuje hned několik způsobů, jak dosáhnout zrychlení výpočtu. V následujících kapitolách budou představeny metody možnosti zrychlení algoritmu, které byly použity v rámci této práce.

Ještě před uvedením jednotlivých metod je nutné podotknout, že veškeré výpočty algoritmu byly provedeny s využitím programu Python 3.7 ve vývojovém prostředí PyCharm. Výpočetní program běžel na osobním počítači Dell Inspiron 15 5510 s procesorem Intel Core i5 11. generace s frekvencí 3,10 GHz.

4.1 Nastavení limit pro výpočet pouze s přípustnými hodnotami

První možností, jak snížit výpočetní čas algoritmu, je zavedení limit pro výpočet pouze s přípustnými, nebo relevantními hodnotami. To je z toho důvodu, že základní algoritmus provádí výpočet pro všechny stavové a řídicí hodnoty, i ty které v daný časový krok nejsou reálné, nebo relevantní. Jedná se tedy převážně o vhodné nastavení horní a spodní hranice stavu nabití baterie pro zvolený cyklus a počáteční podmínku a nastavení limit pro přípustnou řídicí hodnotu, tedy točivý moment elektromotoru, v každém časovém kroku výpočtu.



Pro stav nabití baterie se tedy vhodně nastaví horní a spodní hranice SOC na základě jízdního cyklu a zvolené hodnoty stavu nabití baterie po ujetí trasy. Díky tomu se zmenší interval SOC, který je třeba diskretizovat, a zmenší se tak i počet stavových hodnot, a tudíž i počet iterací při výpočetním procesu. Limitní podmínka je dána následujícím vztahem:

$$SOC_{low} \leq SOC \leq SOC_{high} \quad (68)$$

Podobný princip platí i u točivého momentu elektromotoru. V každém časovém okamžiku se nastaví limity pro výběr řídicí hodnoty, čímž se opět sníží počet iterací při výpočetním procesu. Jedná se hlavně o vyřazení hodnot točivého momentu elektromotoru, pro které v daný okamžik výpočet nemá smysl, nebo je nereálný. Limity jsou dány následujícími vztahy:

$$M_{EM,lim\ min}(t) = \min \left(\max \left((M_{req}(t) - M_{SM\ av}(t)), M_{EM\ min}(t) \right), M_{EM\ max}(t) \right) \quad (69)$$

Kde tento vztah vyjadřuje limitu minima. $M_{SM\ av}$ je dostupný točivý moment spalovacího motoru. $M_{EM\ min}$ a $M_{EM\ max}$ jsou limitní hodnoty elektromotoru a baterie, co se týče točivého momentu.

Pro limitu maxima platí následující vztah:

$$M_{EM,lim\ max}(t) = \max \left(\min(M_{req}(t), M_{EM\ max}(t)), M_{EM\ min}(t), 0 \right) \quad (70)$$

Při zavedení limit pro výpočet pouze s přípustnými, nebo relevantními hodnotami se dosáhlo průměrného zrychlení výpočtu o 61%. V tabulce č. 5 je vidět porovnání výpočetních časů pro jednotlivé jízdní cykly.

Jízdní cyklus	NEDC	WLTC	FTP 75	FTP HW
Zrychlení	61,7 %	62,7 %	63,6 %	59,3 %
Výpočetní čas	345 s → 132 s	594 s → 221 s	593 s → 215 s	300 s → 122 s

Tabulka 5: Porovnání výpočetních časů po zavedení limit pro výpočet pouze s přípustnými hodnotami

Všechny zbylé porovnání jsou s výpočtem s nastavenými limity z důvodu porovnatelnosti spotřeby paliva a reálnosti výsledku



4.2 Memorizace

Dalším způsobem zrychlení výpočtu algoritmu je metoda zvaná memorizace (anglicky memoization). Memorizace je optimalizační technika používaná především k urychlení počítačových programů, čehož se dosáhne pomocí ukládání výsledků výpočetně náročných funkcí do paměti počítače. Když se pak objeví stejné vstupní hodnoty, tak se místo opětovného výpočtu vezme již uložený výsledek z paměti. Memorizace je způsob, jak snížit časové náklady výpočtu výměnou za paměťové náklady. To znamená, že pamatováním výsledků se optimalizuje rychlost výpočtu výměnou za vyšší využití místa v paměti počítače. Hlavní výhodou této metody je fakt, že se jedná pouze o způsob zrychlení algoritmu, který na přesnost výpočtu nemá žádný vliv. [30] [31]

Pomocí memorizace se ukládají výsledky do paměti na základě specifických vstupů. Funkci lze uložit do paměti pouze v případě, že volání funkce má stejný efekt jako nahrazení této funkce pomocí volání výsledné hodnoty z paměti. Memorizace většinou naplňuje svou mezipaměť výsledků transparentně za běhu, ale je možné naplnit paměť i předem a memorizace pak pracuje podobně jako look-up table. [30]

V rámci této práce se metoda memorizace použila pro ukládání výsledů při výpočtu cenové funkce $J_{N-K,N}(x^i(N-K), u^j(N-K))$ a změny stavu nabití baterie za jeden časový krok ΔSOC , které jsou součástí rekurzivního výpočtu a je nutné je počítat v každé iteraci. Na základě vstupních hodnot, kterými jsou úhlová rychlost $\omega(N-K)$ a točivý moment elektromotoru, tedy řídicí hodnota $u^j(N-K)$, se výsledky cenové funkce a změny stavu nabití baterie ukládají do paměti. Jestliže budou při dalších iteracích stejné vstupní hodnoty, už není nutné znovu provádět výpočet, ale pouze se vezme výsledek z paměti.

Při použití metody memorizace se dosáhlo průměrného zrychlení výpočtu o 71,7%. V tabulce č. 6 je vidět porovnání výpočetních časů pro jednotlivé jízdní cykly.

Jízdní cyklus	NEDC	WLTC	FTP 75	FTP HW
Zrychlení	72,4 %	70 %	70,1 %	74,3 %
Výpočetní čas	132 s → 36,5 s	221 s → 66,4 s	215 s → 64,5 s	122 s → 31,4 s

Tabulka 6: Porovnání výpočetních časů při použití metody memorizace



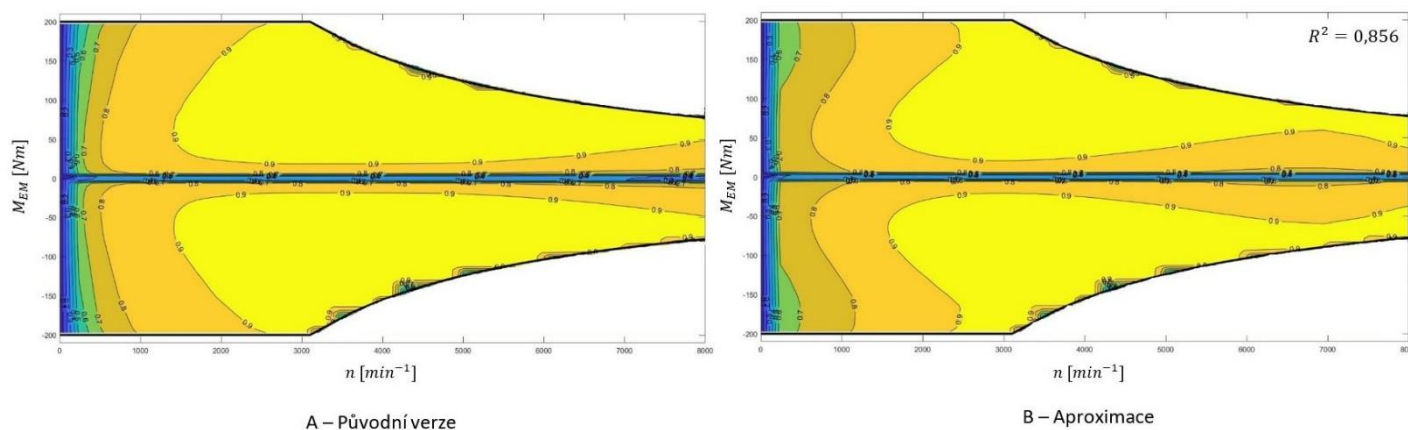
4.3 Nahrazení map účinností pomocí aproximace polynomem

Co se týče výpočetní rychlosti algoritmu, tak jeden z hlavních problémů představují mapy účinností spalovacího motoru a elektromotoru ve formě look-up table, které se využívají pro výpočty pro každou řídicí hodnotu v každém časovém kroku. Při každém tomto výpočtu je nutné získat hodnotu účinnosti, nebo měrné spotřeby paliva pomocí bilineární interpolace. To samozřejmě není z pohledu výpočetní náročnosti úplně ideální. Z tohoto důvodu se mapy účinnosti v podobě look-up table nahradí aproximací polynomem. Hodnoty z map se pak budou získávat výpočtem rovnice s koeficienty v závislosti na točivém momentu spalovacího motoru, nebo elektromotoru a úhlové rychlosti, protože to je výpočetně mnohem výhodnější. [32] [33] [34]

Pro nahrazení mapy účinnosti elektromotoru byl zvolen polynom třetího stupně pro každý směr. Výpočet je pak dán následujícím vztahem:

$$f(x, y) = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + a_{30}x^3 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3 \quad (71)$$

Kde x představuje úhlovou rychlost, y je točivý moment elektromotoru a a_{ij} jsou specifické koeficienty unikátní pro danou mapu, které byly získány pomocí zabudované funkce v programu MATLAB. Porovnání původní a aproximované mapy je možné vidět na obrázku č. 18. Součástí aproximované mapy je i koeficient R^2 vyjadřující kvalitu aproximace. [34]



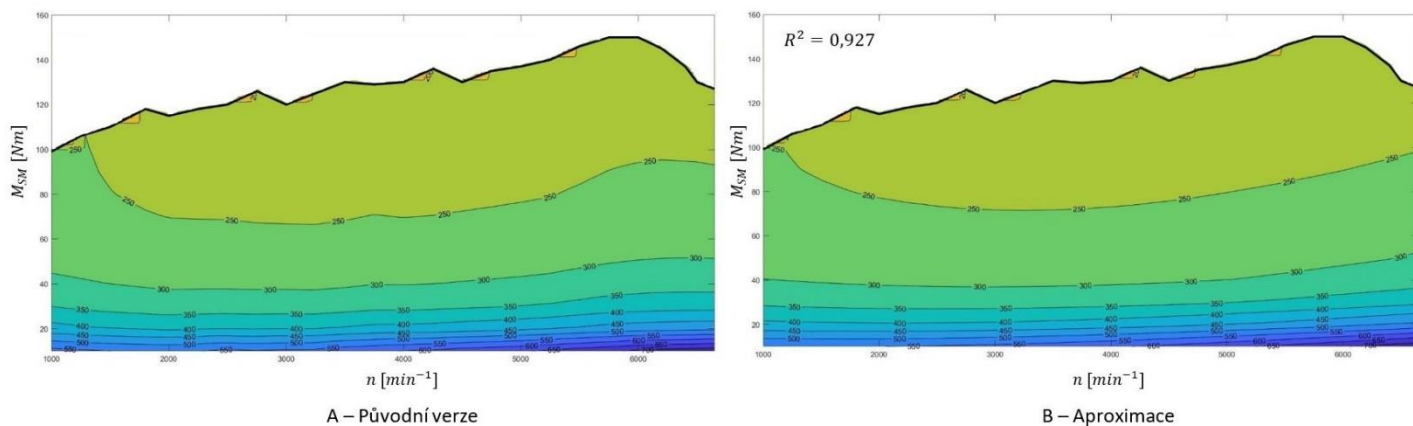
Obrázek 18: Porovnání mapy účinnosti elektromotoru s aproximací

Pro nahrazení mapy pro spalovací motor byla zvolena vyšší přesnost, protože hraje ve výpočtu větší roli. Pro nahrazení byl zvolen polynom druhého stupně ve směru x a polynom pátého stupně ve směru y . Výpočet je pak dán vztahem (72).



$$f(x, y) = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3 + a_{22}x^2y^2 + a_{13}xy^3 + a_{04}y^4 + a_{23}x^2y^3 + a_{14}xy^4 + a_{05}y^5 \quad (72)$$

Kde x opět představuje úhlovou rychlost, y je točivý moment spalovacího motoru. Porovnání původní a aproximované mapy je možné vidět na obrázku č. 19.



Obrázek 19: Porovnání mapy spalovacího motoru s aproximací

Při použití aproximace polynomem se dosáhlo průměrného zrychlení výpočtu o 63,4% a průměrný rozdíl spotřeby oproti výpočtu s původními mapami je 1,35%. V tabulce č. 7 je vidět porovnání výpočetních časů pro jednotlivé jízdní cykly.

Jízdní cyklus	NEDC	WLTC	FTP 75	FTP HW
Zrychlení	62,4 %	60,1 %	62,6 %	68,5 %
Výpočetní čas	132 s → 49,7 s	221 s → 88,1 s	215 s → 80,6 s	122 s → 38,4 s
Rozdíl spotřeby	0,54 %	2,35 %	1,37 %	1,13 %

Tabulka 7: Porovnání výpočetních časů při použití aproximace polynomem

V tabulce č. 8 je ještě vidět porovnání výpočetních časů při použití jak metody memorizace, tak aproximace map polynomem. Oproti základnímu algoritmu je výpočet s použitím všech metod zrychlení průměrně 10,2x rychlejší.

Jízdní cyklus	NEDC	WLTC	FTP 75	FTP HW
Celkové zrychlení	74,6 %	72,7 %	72,4 %	77,2 %
Výpočetní čas	132 s → 33,6 s	221 s → 60,4 s	215 s → 59,53 s	122 s → 27,8 s

Tabulka 8: Porovnání výpočetních časů při použití memorizace i aproximace polynomem



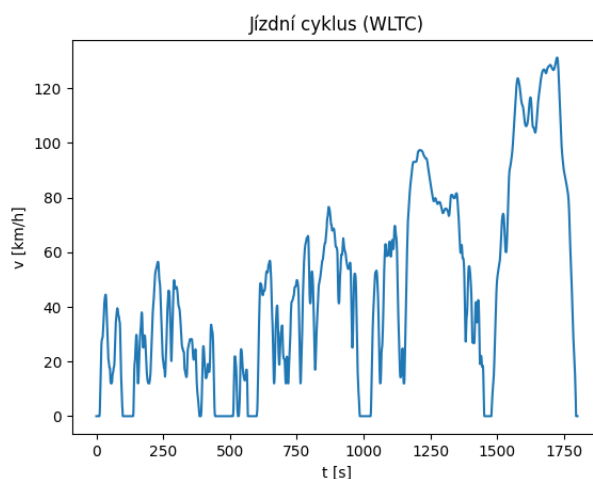
5 Výsledky simulace

V této kapitole budou ukázány výsledky simulací po projetí jednotlivých jízdních cyklů s různými počátečními podmínkami. Pro všechny simulace byly zvoleny stejné hodnoty diskretizace stavových a řídicích hodnot – $dSOC = 0,1\%$ a $dM_{EM} = 2 Nm$. Zároveň jsou při výpočtech použity všechny metody optimalizace výpočetního času uvedené v předchozí kapitole. Detailnější ukázka všech počítaných parametrů v simulaci bude ukázána pro jeden jízdní cyklus a počáteční podmínku a pro zbytek budou prezentovány pouze nejdůležitější parametry, které nás z hlediska řízení hybridního pohonu zajímají – tedy hlavně průběh stavu nabití baterie a spotřeba paliva a elektrické energie.

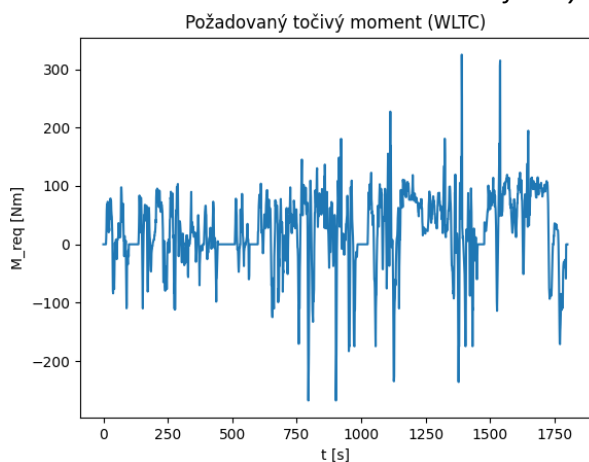
Cyklus WLTC

Detailnější ukázka všech počítaných parametrů v simulaci bude prezentována pro jízdní cyklus WLTC, a to při zadání počáteční podmínky, že se po ujetí jízdního cyklu sníží stav nabití baterie o 5%.

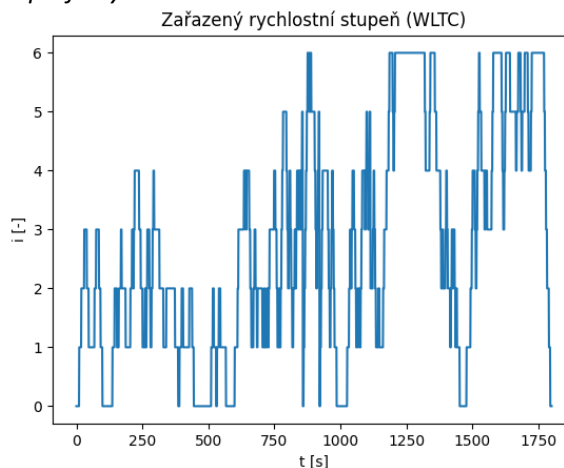
První tři grafy jsou stejné pro všechny počáteční podmínky, protože závisí pouze na vstupních parametrech z rychlostního profilu a parametrech vozidla.



Graf 7: Rychlostní profil cyklu WLTC



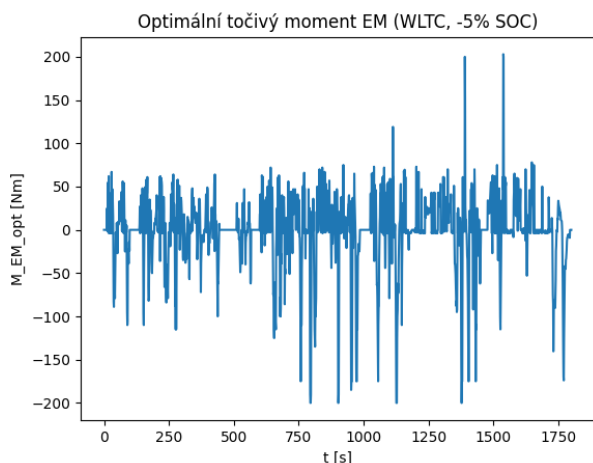
Graf 8: Požadovaný točivý moment



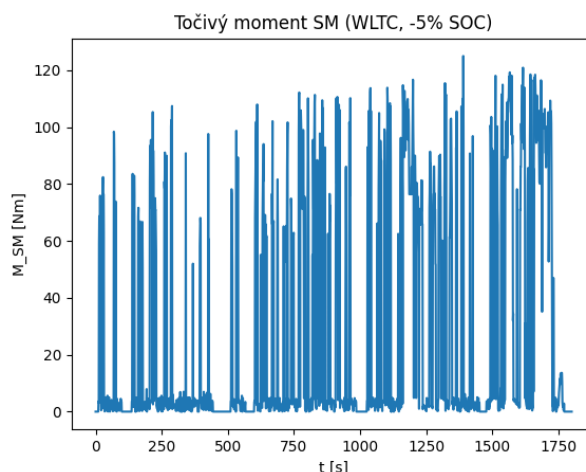
Graf 9: Zařazený rychlostní stupeň



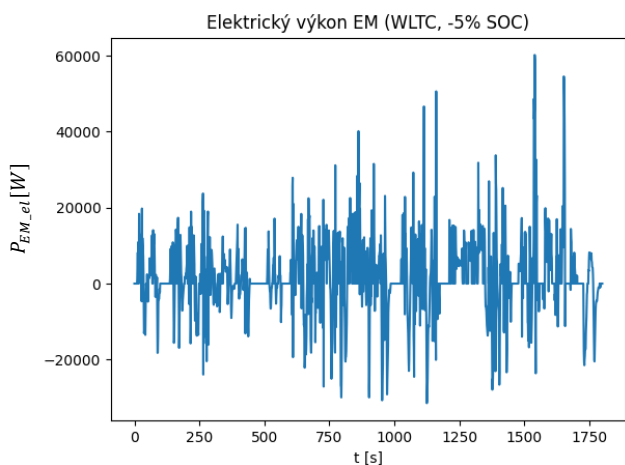
Zbylé grafy jsou podmíněné výpočtem algoritmu dynamického programování a závisí na zvolených počátečních podmínkách. Jeden z důležitějších parametrů je optimální točivý moment elektromotoru, který představuje optimální řídicí hodnotu pro dosažení globálně optimálního řešení. Jeho průběh je vidět na grafu č. 10.



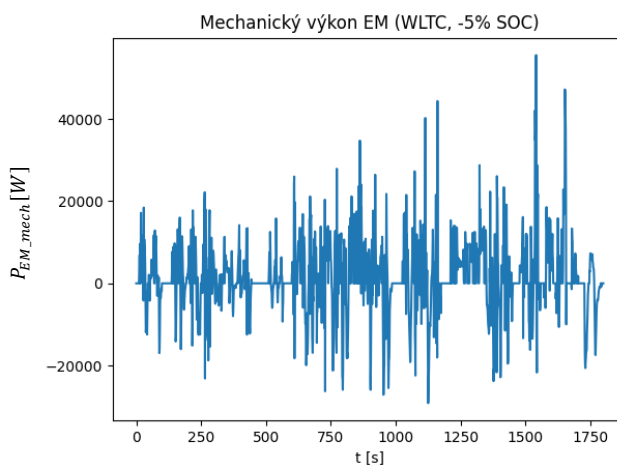
Graf 10: Optimální točivý moment elektromotoru



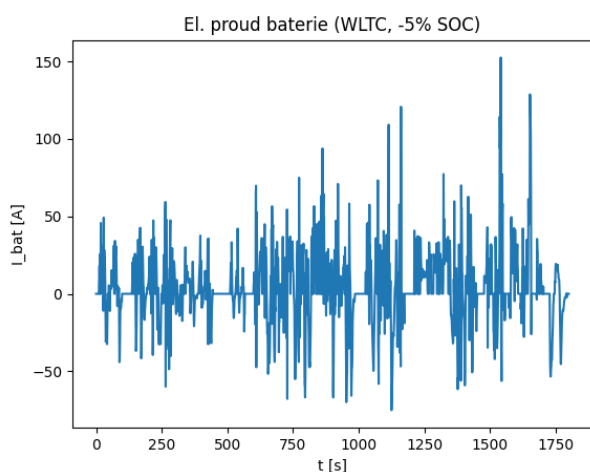
Graf 11: Točivý moment spalovací motoru



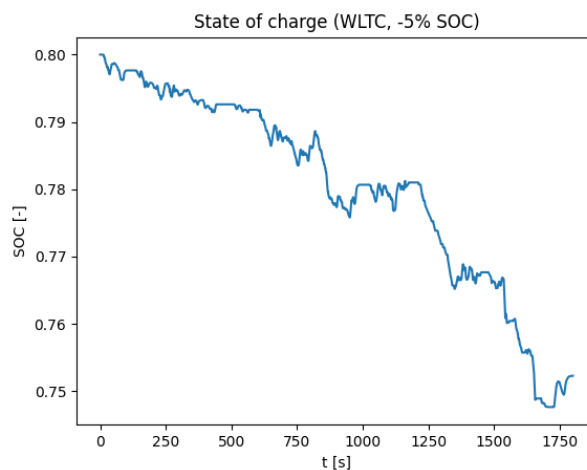
Graf 12: Elektrický výkon elektromotoru



Graf 13: Mechanický výkon elektromotoru



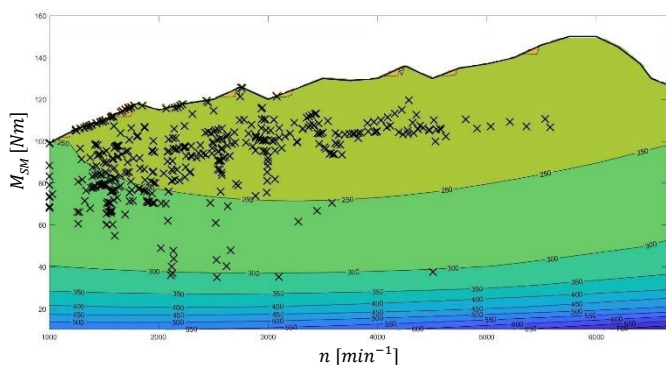
Graf 14: Elektrický proud baterie



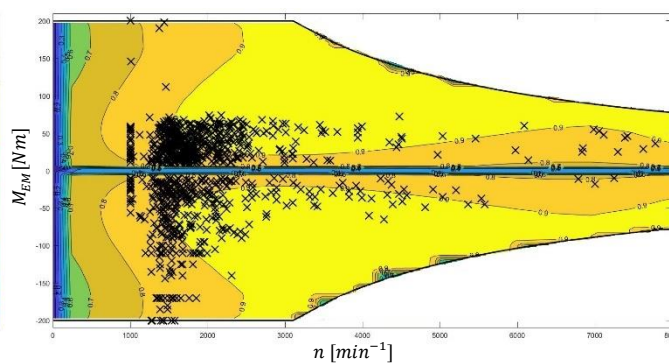
Graf 15: Stav nabití baterie



Na následujících dvou grafech jsou vidět pracovní body spalovacího motoru a elektromotoru. Podle grafu č. 16 je vidět, že spalovací motor pracuje převážně v oblasti s vyšší účinností, což je také jedním z cílů strategie řízení hybridního pohonu.

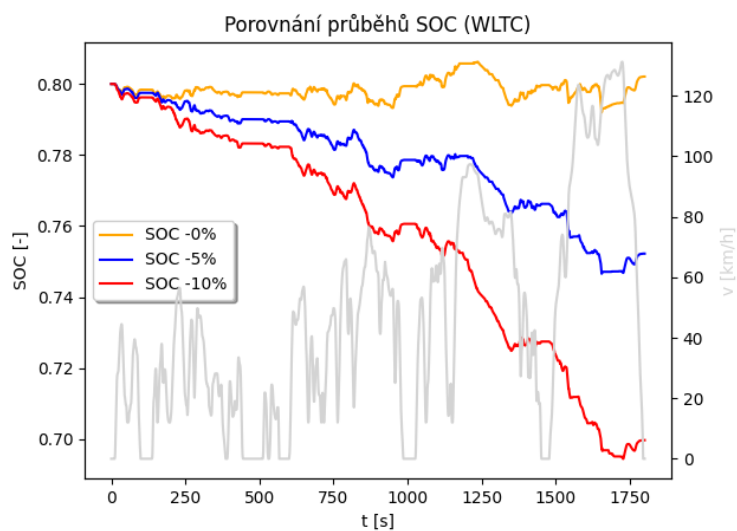


Graf 16: Pracovní body spalovacího motoru



Graf 17: Pracovní body elektromotoru

Na posledním grafu je ještě vyobrazeno porovnání průběhů stavu nabití baterie v závislosti na zvolené počáteční podmínce. Je zde zobrazen průběh pro charge – sustaining mode (SOC -0%) a potom snížení stavu nabití baterie po ujetí jízdního cyklu o 5% a 10%.

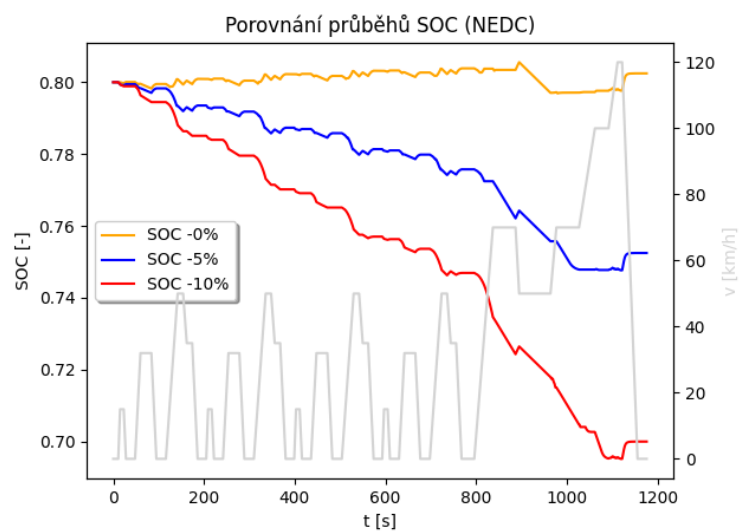


Graf 18: Porovnání průběhů SOC v závislosti na počáteční podmínce (WLTC)

Porovnání kombinované spotřeby pro jednotlivé jízdní cykly je uvedené na konci této kapitoly v tabulce č. 9.

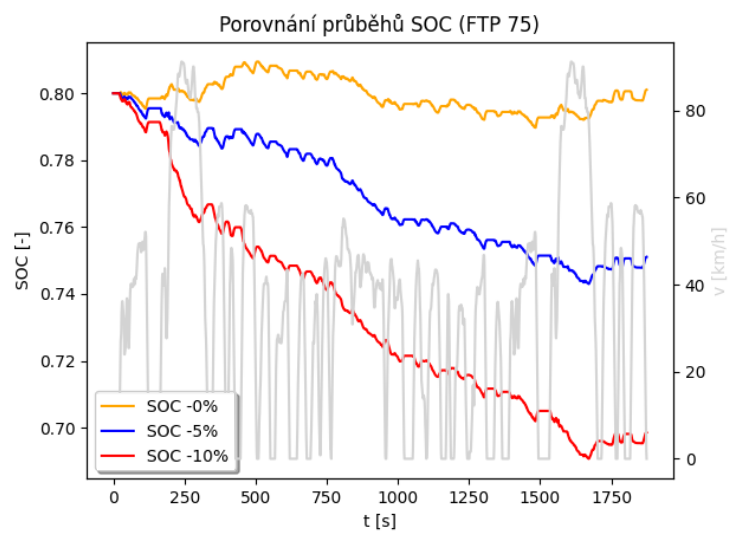


Cyklus NEDC



Graf 19: Porovnání průběhů SOC v závislosti na počáteční podmínce (NEDC)

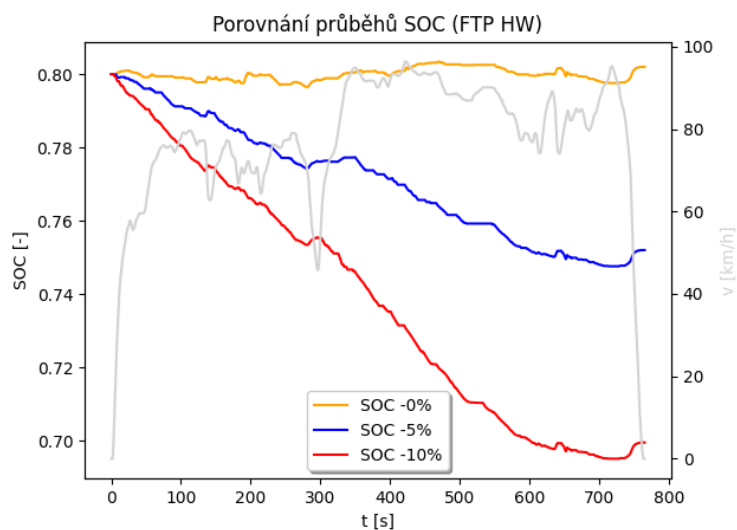
Cyklus FTP 75



Graf 20: Porovnání průběhů SOC v závislosti na počáteční podmínce (FTP 75)



Cyklus FTP HW



Graf 21: Porovnání průběhů SOC v závislosti na počáteční podmínce (FTP HW)

Jízdní cyklus		Spotřeba paliva [l/100km] / Spotřeba elektrické energie [kWh/100km]							
		WLTC		NEDC		FTP 75		FTP HW	
Konvenční vozidlo		7,41		6,36		6,51		5,61	
Hybrid	Charge – sustaining mode	5,21	5,97	5,15	5,15	4,44	7,73	5,25	2,45
	SOC -5%	4,22	8,18	3,04	10,66	3,21	11,07	3,86	5,88
	SOC -10%	3,27	11,04	1,77	14,72	1,98	14,11	2,54	10,56

Tabulka 9: Porovnání spotřeby paliva a elektrické energie pro jednotlivé jízdní cykly



Shrnutí

Pro odladění výpočetního procesu a pro lepší porozumění, jak hodnoty nastavovaných parametrů ovlivní výpočet a výsledek simulace, byla provedena citlivostní analýza. Citlivostní analýza byla provedena pro tři hlavní parametry, které je možné v simulaci přímo nastavit, nebo měnit. Jedná se o velikost diskretizace stavu nabití baterie, diskretizace točivého momentu elektromotoru a nastavení počáteční podmínky SOC po ujetí jízdního cyklu.

Analýza citlivosti na diskretizaci SOC byla provedena pro kroky od 0,05% až do 1% SOC. Výpočetní náročnost se zvyšuje téměř lineárně s použitím menších kroků. Co se týče přesnosti výpočtu, což je možné ověřit například hodnotou spotřeby paliva, tak použitím větších kroků se do výpočtu vnáší poměrně velké nepřesnosti. Při použití kroku 0,25% a menších už jsou výsledky téměř stejné.

Analýza diskretizace točivého momentu elektromotoru byla provedena pro kroky od $1Nm$ do $10Nm$. Výpočetní čas má opět lineární závislost a hodnota spotřeby paliva je téměř stejná pro všechny zkoušené kroky. Z toho je možné usoudit, že velikost diskretizace točivého momentu nemá na výpočet takový vliv jako diskretizace stavu nabití baterie. Nicméně pro jistotu dosažení dostatečně přesných výsledků je vhodné použít hodnotu diskretizace alespoň $5Nm$, ideálně menší.

Poslední analýza byla provedena pro různé hodnoty počáteční podmínky SOC po ujetí jízdního cyklu. Různé počáteční podmínky budou mít samozřejmě vliv na hodnotu spotřeby paliva, protože přímo souvisí s tím, v jaké míře bude při jízdě využíván elektromotor a tím pádem i spalovací motor. Zajímavým zjištěním byl ale fakt, že hodnota počáteční podmínky má vliv i na výpočetní čas. Se zvětšujícím se rozdílem mezi počáteční a koncovou hodnotou SOC se zvyšuje i výpočetní čas algoritmu. Poslední ovlivnění výpočtu spojené s hodnotou počáteční podmínky, které bylo zmíněno už v kapitole 4.4.1, je možnost nastavení limit pro diskretizaci stavu nabití baterie. Z výsledků simulace je zřejmé, že hodnota SOC se bude v průběhu simulace pohybovat téměř výhradně mezi počáteční a koncovou hodnotou SOC (pokud se nejedná o charge-sustaining mode). Z tohoto důvodu je výhodné nastavit limity pro diskretizaci stavu nabití baterie v intervalu 5-10 % od těchto hodnot, aby se výpočet neprováděl pro zbytečně velké množství stavových hodnot.



6 Porovnání vyvinutého algoritmu s implementací dynamického programování v programu GT-Suite

Software GT-Suite je simulační nástroj s různými funkcemi a knihovnami zaměřenými na širokou škálu aplikací a odvětví. Program je velmi často využíván i v automobilovém průmyslu. Inženýrům nabízí funkce od rychlého návrhu konceptu až po podrobné analýzy systémů nebo komponent, optimalizaci návrhu a vyšetřování příčin problému. Jednou z funkcí GT-Suite je i optimalizace hybridního pohonného systému pomocí různých metod zabudovaných přímo v programu. Jednou z těchto metod je i optimalizace s implementací dynamického programování, a proto bude následovat v této kapitole porovnání vyvinutého algoritmu s implementací dynamického programování v programu GT-Suite.

6.1 Princip výpočetního procesu

Implementace dynamického programování v rámci programu GT-Suite pracuje na podobné bázi jako princip vysvětlený v kapitole 3.4.2, který byl použitý pro vývoj řídicího algoritmu v této práci. Jsou zde ale drobné rozdíly jak už přímo ve výpočtu, tak například i ve značení. Hlavním rozdílem je použití tzv. penalizačních funkcí. Princip výpočetního procesu vychází z [44] a v této sekci bude nastíněn stručný popis.

Optimalizační úloha i v tomto případě spočívá v minimalizaci cenové funkce J pomocí rekurzivního výpočtu. Výpočet cenové funkce je dán následujícím vztahem:

$$J_{\pi}(x_0) = g_N(x_N) + T_N(x_N) + \sum_{k=0}^{N-1} L_k(x_k, u_k(x_k)) + p_k(x_k) \quad (73)$$

Kde x a u jsou opět stavové a řídicí hodnoty, hodnota $g_N(x_N)$ představuje cenu finálního stavu a $T_N(x_N)$ je dodatečná penalizace finálního stavu, která částečně omezuje konečný stav. Člen $L_k(x_k, u_k(x_k))$ představuje cenu operace při použití řídicí hodnoty u_k ve stavu x_k . Posledním členem je $p_k(x_k)$, což je již zmiňovaná penalizační funkce. [44] [45]

Výpočet hodnoty $T_N(x_N)$ je dán vztahem (74), kde v rovnici figurují ještě váhový faktor γ a exponent penalizace finálního stavu β . SOC_{grid} představuje diskretizovaný stav nabití baterie a SOC_{target} je zvolená konečná hodnota stavu nabití baterie.

$$T_N = \gamma(SOC_{grid} - SOC_{target})^{\beta} \quad (74)$$



Penalizační funkce $p_k(x_k)$ zavádí v platnost stavová omezení pro $k = 0, 1, \dots, N - 1$. Výpočet této funkce je pak dán rovnicí (75), kde figuruje také váhový faktor λ a exponent penalizační funkce α . SOC_{min} a SOC_{max} jsou limitní hodnoty pro stav nabití baterie.

$$p(SOC) = \lambda \left(\frac{SOC(t) - SOC_{target}}{\frac{SOC_{max} - SOC_{min}}{2}} \right)^\alpha \quad (75)$$

Optimální řízení je pak takové, které minimalizuje hodnotu $J_\pi(x_0)$ ze vztahu (73).

$$J^*(x_0) = \min_{x \in \pi} J_\pi(x_0) \quad (76)$$

Dynamické programování na základně principu optimality vyhodnotí optimální cenovou funkci $J_\pi(x^i)$ pro každý diskretizovaný stav pomocí rekurzivního výpočtu. Výpočet pro konečný stav je znázorněn vztahem (77) a výpočet cenové funkce pro kroky $k = 0, 1, \dots, N - 1$ je uveden ve vztahu (78).

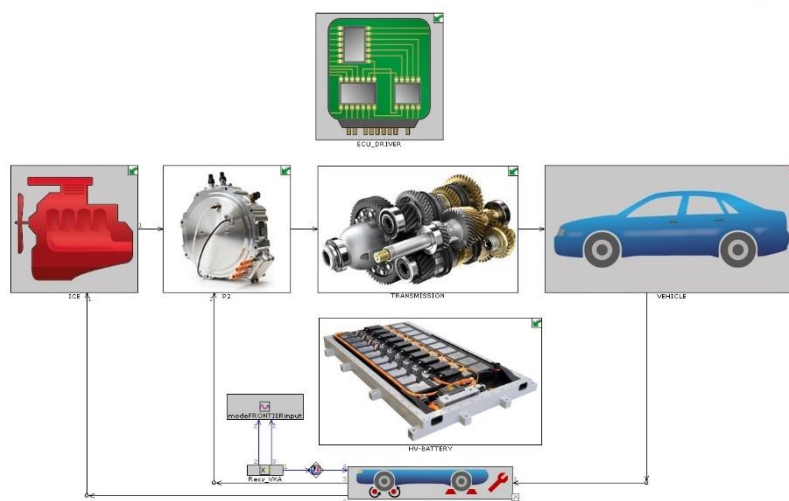
$$J_N(x^i) = g_N(x^i) + T_N(x^i) \quad (77)$$

$$J_k(x^i) = \min_{u_k \in U_k} \{L_k(x^i, u_k) + p_k(x_k) + \dots + J_{k+1}(f_k(x^i, u_k))\} \quad (78)$$

Pravá strana rovnice (78) je minimalizována v každém diskretizovaném stavu pro dosažení optimálního řízení. [44] [45]

6.2 Simulace

Simulace probíhá na základě modelu hybridního vozidla, kde parametry modelu byly zvoleny stejné jako pro simulaci v programu Python. Model je ale mnohem komplexnější, takže zde byly použity i parametry, které byly při návrhu algoritmu v rámci této práce zanedbány. Na obrázku č. 20 je vidět blokové schéma modelu vozidla v prostředí programu GT-Suite, kde každý blok v sobě ukrývá ještě další subsystém.



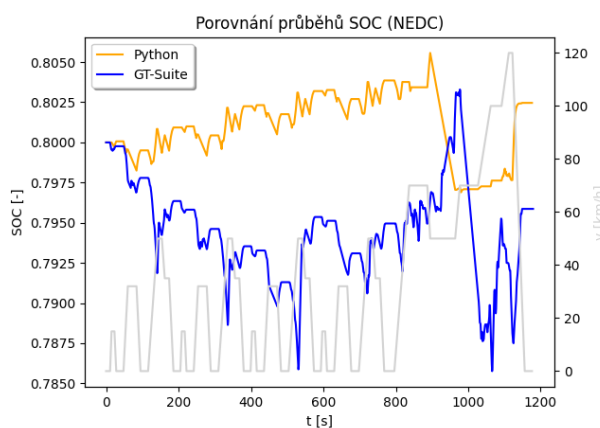
Obrázek 20: Blokové schéma modelu v prostředí GT-Suite



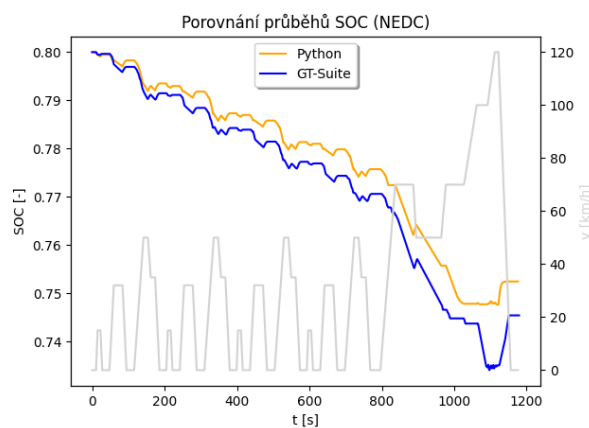
Na následujících grafech je možné vidět porovnání výsledků simulace vyvinutého algoritmu a implementace dynamického programování v GT-Suite. Jedná se o porovnání průběhů stavu nabití baterie pro charge-sustaining mode a pro počáteční podmínky, že se po ujetí jízdního cyklu sníží stav nabití baterie o 5% a o 10%. Na základě grafického porovnání obou simulací je možné vidět malé odlišnosti. To je nejspíše způsobeno drobnými rozdíly ve výpočetním procesu, jako například použití penalizačních funkcí, nebo větší komplexností modelu v programu GT-Suite. To se týká například modelu baterie, kde některé parametry byly ve vyvinutém algoritmu pro zjednodušení nahrazeny konstantami, zatímco v programu GT-Suite jsou definovány pomocí look-up table.

Na grafu č. 22 je porovnání stavu nabití baterie pro charge-sustaining mode při simulaci na jízdním cyklu NEDC. Už na první pohled je vidět rozdíl v průběhu SOC pro jednotlivé simulace. Zatímco u vyvinutého algoritmu se SOC v první části jízdního cyklu udržuje kolem počáteční hodnoty, nebo dochází k mírnému dobíjení, tak průběh ze simulace v GT-Suite v první polovině cyklu výrazněji klesá a potom dochází k nabíjení baterie před poslední částí cyklu, kde je potřeba největší výkon. Na grafech č. 23 a 24 je zase vidět, že se průběhy SOC jednotlivých simulací ve velké míře vzájemně kopírují, akorát je mezi nimi mírný posun.

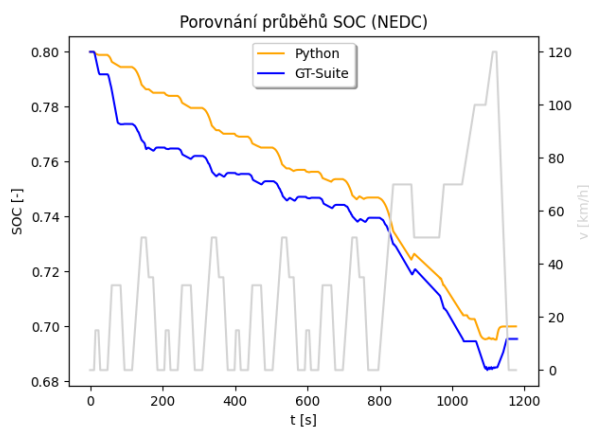
Na přesnější pochopení rozdílů mezi jednotlivými simulacemi by bylo vhodné vytvořit citlivostní analýzu na parametry dynamického programování v programu GT-Suite, ale to už je bohužel nad rámec této práce.



Graf 22: Porovnání průběhů stavu nabití baterie pro charge-sustaining mode



Graf 23: Porovnání průběhů SOC s počáteční podmínkou úbytku 5%



Graf 24: Porovnání průběhů SOC s počáteční podmínkou úbytku 10%

V tabulce č. 10 je ještě zobrazeno porovnání spotřeby paliva pro obě simulace. Z obou simulací vycházejí podobné hodnoty, kdy rozdíly ve spotřebě jsou v rámci několika procent.

	Spotřeba paliva [l/100km]	
	Simulace Python	Simulace GT-Suite
Charge-sustaining mode	5,15	5,06
SOC -5%	3,04	3,16
SOC -10%	1,77	1,94

Tabulka 10: Porovnání spotřeby paliva simulace v programu Python a GT-Suite



Závěr

Cílem diplomové práce bylo navrhnout a optimalizovat globálně optimální algoritmus pro řízení hybridního pohonného systému s využitím dynamického programování a simulovat jeho funkci v programu Python. Práce byla rozdělena na tři hlavní části.

První částí byla teoretická část práce, která byla zaměřená na rešeršní zpracování hybridních vozidel a stávajících strategií řízení hybridních pohonů. Nejdříve tedy byla vysvětlena problematika hybridních vozidel, především co se týká topologie a stupně elektrifikace. Poté byly představeny aktuálně používané řídicí strategie. V současnosti se používají heuristické metody, nebo optimalizační metody řízení, přičemž pomocí optimalizačních metod je možné dosáhnout lepších výsledků. Jednou z nich je i dynamické programování, kterým se tato práce z velké části zabývala.

Druhou částí byl samotný vývoj a optimalizace řídicího algoritmu. Algoritmus byl navržen pro paralelní hybrid v konfiguraci P2. Jednou z hlavních funkcí strategie řízení je optimální rozdělení požadovaného točivého momentu mezi spalovací motor a elektromotor pro dosažení maximálního potenciálu hybridního pohonu. Pro simulaci byly zvoleny čtyři jízdní cykly a na základě vstupních hodnot z těchto cyklů a parametrů vozidla se určí požadovaný točivý moment, který se pomocí algoritmu s využitím dynamického programování rozdělí mezi spalovací motor a elektromotor, pro dosažení úspory paliva. Dále byla provedena optimalizace základního algoritmu z hlediska zvýšení výpočetní rychlosti. Byly použity tři způsoby, díky nimž bylo dosaženo znatelného zrychlení výpočtu se zachováním dostatečné přesnosti. Nakonec byla provedena citlivostní analýza na parametry dynamického programování.

Poslední částí pak bylo porovnání vyvinutého algoritmu s implementací dynamického programování v programu GT-Suite. Pro simulaci v programu GT-Suite se použily stejné parametry vozidla a všech komponent pohonného systému. Výpočetní proces dynamického programování v softwaru GT-Suite má několik odlišností od algoritmu použitého v této práci a model vozidla je mnohem komplexnější. Z těchto důvodů je při porovnání výsledků obou způsobů simulací možné vidět malé rozdíly.

Byly splněny všechny cíle, které byly pro diplomovou práci stanoveny, a to počínaje rešeršním zpracováním problematiky strategie řízení hybridních pohonů až po samotný vývoj a optimalizaci algoritmu s použitím dynamického programování a následnou simulaci v programu Python. Vytvořený skript pro simulaci algoritmu řízení je součástí přílohy diplomové práce.



Seznam použitých informačních zdrojů

- [1] GUZZELLA, L. a A. SCIARRETTA. *Vehicle Propulsion Systems*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2013. ISBN 978-3-642-35912-5.
- [2] *Hybrid Electrical Vehicles* [online]. [cit. 2022-04-27]. Dostupné z: http://www.ae.pwr.wroc.pl/filez/20110606092353_HEV.pdf
- [3] Vidyanandan, K.V.. (2018). Overview of Electric and Hybrid Vehicles. Energy Scan (A House Journal of Corporate Planning, NTPC Ltd., India). III. 7-14.
- [4] JOHASSON, Karin. *Analysing Hybrid Drive System Topologies*. Sweden: Media-Tryck, 2002. ISBN 91-88934-23-3.
- [5] EMADI, Ali a Kaushik RAJASHEKARA. *Topological Overview of Hybrid Electric and Fuel Cell Vehicular Power System Architectures and Configurations*.
- [6] Athani, G., Dongare, K., Balusu, R., Gupta, S. et al., "A Novel Approach to Enhance Stop/Start Battery Life in a Vehicle with Micro Hybrid System Functions," SAE Technical Paper 2016-01-0007, 2016, doi:10.4271/2016-01-0007.
- [7] Lee, S., Cherry, J., Safoutin, M., McDonald, J. et al., "Modeling and Validation of 48V Mild Hybrid Lithium-Ion Battery Pack," SAE Int. J. Alt. Power. 7(3):273–287, 2018, doi:10.4271/2018-01-0433.
- [8] Yamagishi, T. and Ishikura, T., "Development of Electric Powertrain for CLARITY PLUG-IN HYBRID," SAE Int. J. Alt. Power. 7(3):323–333, 2018, doi:10.4271/2018-01-0415.
- [9] Malmir, F., Xu, B., and Filipi, Z., "A Heuristic Supervisory Controller for a 48V Hybrid Electric Vehicle Considering Fuel Economy and Battery Aging," SAE Technical Paper 2019-01-0079, 2019, doi:10.4271/2019-01-0079.
- [10] KIRK, Donald E. *Optimal control theory: an introduction*. Dover edition. Mineola: Dover Publications, 2004. ISBN 978-0486434841.
- [11] Lodaya, D., Zeman, J., Okarmus, M., Mohon, S. et al., "Optimization of Fuel Economy Using Optimal Controls on Regulatory and Real-World Driving Cycles," SAE Int. J. Advances & Curr. Prac. in Mobility 2(3):1705-1716, 2020, doi:10.4271/2020-01-1007.
- [12] Průběh SOC plug-in hybridního vozidla [online]. In: . [cit. 2022-06-16]. Dostupné z: https://www.mdpi.com/energies/energies-10-00412/article_deploy/html/images/energies-10-00412-g001.png
- [13] Bellmanova rovnice. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2022-06-18]. Dostupné z: https://cs.wikipedia.org/wiki/Bellmanova_rovnice
- [14] Series HEV drivetrain [online]. In: . [cit. 2022-06-20]. Dostupné z: <https://www.researchgate.net/profile/Teddy-Gunawan/publication/235639375/figure/fig8/AS:647512042000394@1531390309586/Series-HEV-drive-train.png>
- [15] Series-Parallel HEV drivetrain [online]. In: . [cit. 2022-06-20]. Dostupné z: <https://www.researchgate.net/profile/Gabriele-Perozzi/publication/326655390/figure/fig3/AS:653032178073603@1532706412861/HEV-Series-Parallel-drivetrain-configuration.png>



- [16] Bauer, K. and Gauterin, F., "A Two-Layer Approach for Predictive Optimal Cruise Control," SAE Technical Paper 2016-01-0634, 2016, doi:10.4271/2016-01-0634.
- [17] Aerodynamika vozidla [online]. In: . [cit. 2020-06-03]. Dostupné z: https://encrypted-tbn0.gstatic.com/images?q=tbn%3AAND9GcTOXV2If5IN6mKenA6_-FWM1dle0JL4p-o_M6NildgG1gxClBxY&usqp=CAU
- [18] Asynchronní motor. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2020-01-12]. Dostupné z: https://cs.wikipedia.org/wiki/Asynchron%C3%AD_motor
- [19] Elektromotor. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2020-01-12]. Dostupné z: <https://cs.wikipedia.org/wiki/Elektromotor>
- [20] LASOCKI, Jakub. GEARSHIFT STRATEGY IN THE WORLDWIDE HARMONIZED LIGHT VEHICLES TEST PROCEDURE. Warsaw, 2021. Institute of Vehicles, Warsaw University of Technology.
- [21] Worldwide Harmonized Light Vehicles Test Cycle (WLTC) [online]. [cit. 2022-06-27]. Dostupné z: <https://dieselnet.com/standards/cycles/wltp.php>
- [22] New European Driving Cycle. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2022-06-27]. Dostupné z: https://en.wikipedia.org/wiki/New_European_Driving_Cycle
- [23] SHEN, Wenchen a Huilong YU. Optimization of Shift Schedule for Hybrid Electric Vehicle with Automated Manual Transmission [online]. [cit. 2022-07-04].
- [24] Patil, A.A., Motallebiaraghi, F., Meyer, R., and Asher, Z.D., "Comparison of Optimal Energy Management Strategies Using Dynamic Programming, Model Predictive Control, and Constant Velocity Prediction," SAE Technical Paper 2020-01-5071, 2020, doi:10.4271/2020-01-5071.
- [25] WANG, Rui. Dynamic Programming Technique in Hybrid Electric Vehicle Optimization [online]. [cit. 2022-07-04].
- [26] WANG, Wei. Study on Real-Time Control Based on Dynamic Programming for Plug-in Hybrid- Electric Vehicles [online]. [cit. 2022-07-04].
- [27] MALLON, Kevin R. a Francis ASSADIAN. Robustification Through Minimax Dynamic Programming and Its Implication for Hybrid Vehicle Energy Management Strategies [online]. [cit. 2022-07-04].
- [28] YUAN, Zou a Liu TENG. Comparative Study of Dynamic Programming and Pontryagin's Minimum Principle on Energy Management for a Parallel Hybrid Electric Vehicle [online]. [cit. 2022-07-04].
- [29] MORKUS, Josef. Hybridní pohony [přednáška]. Praha: ČVUT v Praze
- [30] Memoization. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2022-07-04]. Dostupné z: <https://en.wikipedia.org/wiki/Memoization>
- [31] Dynamic Programming - Memoization [online]. [cit. 2022-07-04]. Dostupné z: <https://sites.radford.edu/~nokie/classes/360/dp-memoized.html>
- [32] BAJAJ, Chandrajit a Andrew GILLETTE. Polynomial Curves and Surfaces [online]. [cit. 2022-07-05]. Dostupné z: <https://www.cs.utexas.edu/~bajaj/cs384R10/reading/algebraicFEM-chap1.pdf>
- [33] SOUKUP, Jan. Two-level energy management strategy for hybrid electric vehicle using planned-trip information. Praha, 2021. Diplomová práce. ČVUT v Praze.



- [34] Polynomial Models [online]. [cit. 2022-07-05]. Dostupné z: <https://uk.mathworks.com/help/curvefit/polynomial.html#bt9ykh7>
- [35] Goerke, D., Bargende, M., Keller, U., Ruzicka, N. et al., "Optimal Control based Calibration of Rule-Based Energy Management for Parallel Hybrid Electric Vehicles," SAE Int. J. Alt. Power. 4(1):2015, doi:10.4271/2015-01-1220.
- [36] Asher, Z.D., Tunnell, J.A., Baker, D.A., Fitzgerald, R.J. et al., "Enabling Prediction for Optimal Fuel Economy Vehicle Control," SAE Technical Paper 2018-01-1015, 2018, doi:10.4271/2018-01-1015.
- [37] Lee, H., Cha, S., Kim, H., and Kim, S., "Energy Management Strategy of Hybrid Electric Vehicle using Stochastic Dynamic Programming," SAE Technical Paper 2015-01-0019, 2015, doi:10.4271/2015-01-0019.
- [38] WILLIAMS, Kyle. Approximate Stochastic Differential Dynamic Programming for Hybrid Vehicle Energy Management [online]. [cit. 2022-07-06]. Dostupné z: doi:DOI: 10.1115/1.4042253
- [39] ASHER, Zachary D. Real-Time Implementation of Optimal Energy Management in Hybrid Electric Vehicles: Globally Optimal Control of Acceleration Events [online]. [cit. 2022-07-06]. Dostupné z: doi:DOI: 10.1115/1.4046477
- [40] WANG, Ximing. Application Study on the Dynamic Programming Algorithm for Energy Management of Plug-in Hybrid Electric Vehicles [online]. [cit. 2022-07-06]. Dostupné z: doi:doi:10.3390/en8043225
- [41] BIASINI, Riccardo. A near-optimal rule-based energy management strategy for medium duty hybrid truck [online]. [cit. 2022-07-06].
- [42] LEE, Heeyun. A Study on Power Management Strategy of HEV using Dynamic Programming [online]. [cit. 2022-07-06].
- [43] TAGNER, Nikita. Optimal Energy Management for Parallel Hybrid Electric Vehicles using Dynamic Programming [online]. Stockholm, Sweden, 2017 [cit. 2022-07-06]. Dostupné z: www.kth.se/sci. Degree project. KTH ROYAL INSTITUTE OF TECHNOLOGY SCHOOL OF ENGINEERING SCIENCES.
- [44] GAMMA TECHNOLOGIES. Optimization Settings for Backward Kinematic Vehicle State Optimization. 2020.
- [45] TOMAN, Rastislav a Jolana HEŘMANOVÁ. Evaluation of Plug-in Parallel HEV Topologies Using Optimal Control Methods and Vehicle Dynamics Simulation [online]. [cit. 2022-07-08].



Seznam obrázků

Obrázek 1: Sériový hybridní pohon [14]	12
Obrázek 2: Umístění elektromotoru v pohonném systému paralelního hybridu [29]	14
Obrázek 3: Kombinovaný hybrid [15].....	15
Obrázek 4: Stupeň elektrifikace hybridních vozidel	16
Obrázek 5: Blokové schéma řídicího systému hybridního vozidla	21
Obrázek 6: Strategie řízení podle předdefinovaných map [1].....	22
Obrázek 7: Fuzzy logika.....	23
Obrázek 8: Průběh stavu nabití baterie plug-in hybridu [12].....	25
Obrázek 9: Schématické znázornění procesu dynamického programování.....	27
Obrázek 10: Schéma podélné dynamiky vozidla	34
Obrázek 11: Aerodynamika vozidla [17]	35
Obrázek 12: Náhradní schéma baterie [1].....	39
Obrázek 13: Mapa účinnosti elektromotoru	42
Obrázek 14: Úplná charakteristika spalovacího motoru.....	43
Obrázek 15: Vývojový diagram výpočetního procesu algoritmu	54
Obrázek 16: Stavově-časová mřížka.....	55
Obrázek 17: Dopředná rekurze.....	55
Obrázek 18: Porovnání mapy účinnosti elektromotoru s aproximací	63
Obrázek 19: Porovnání mapy spalovacího motoru s aproximací.....	64
Obrázek 20: Blokové schéma modelu v prostředí GT-Suite	72



Seznam grafů

Graf 1: Průběh řazení podle strategie WLTP	45
Graf 2: Rychlostní profil cyklu WLTC	46
Graf 3: Rychlostní profil cyklu NEDC	47
Graf 4: Rychlostní profil cyklu FTP 75.....	47
Graf 5: Rychlostní profil cyklu FTP HW.....	48
Graf 6: Průběhu SOC po simulaci na jízdním cyklu WLTC.....	59
Graf 7: Rychlostní profil cyklu WLTC	65
Graf 8: Požadovaný točivý moment	65
Graf 9: Zařazený rychlostní stupeň.....	65
Graf 10: Optimální točivý moment elektromotoru.....	66
Graf 11: Točivý moment spalovací motoru	66
Graf 12: Elektrický výkon elektromotoru.....	66
Graf 13: Mechanický výkon elektromotoru.....	66
Graf 14: Elektrický proud baterie	66
Graf 15: Stav nabití baterie.....	66
Graf 16: Pracovní body spalovacího motoru	67
Graf 17: Pracovní body elektromotoru	67
Graf 18: Porovnání průběhů SOC v závislosti na počáteční podmínce (WLTC).....	67
Graf 19: Porovnání průběhů SOC v závislosti na počáteční podmínce (NEDC)	68
Graf 20: Porovnání průběhů SOC v závislosti na počáteční podmínce (FTP 75).....	68
Graf 21: Porovnání průběhů SOC v závislosti na počáteční podmínce (FTP HW)	69
Graf 22: Porovnání průběhů stavu nabití baterie pro charge-sustaining mode	73
Graf 23: Porovnání průběhů SOC s počáteční podmínkou úbytku 5%	74
Graf 24: Porovnání průběhů SOC s počáteční podmínkou úbytku 10%	74



Seznam tabulek

Tabulka 1: Vybrané parametry vozidla	33
Tabulka 2: Základní parametry elektromotoru	42
Tabulka 3: Základní parametry spalovacího motoru.....	44
Tabulka 4: Porovnání jízdních cyklů	48
Tabulka 5: Porovnání výpočetních časů po zavedení limit pro výpočet pouze s přípustnými hodnotami.....	61
Tabulka 6: Porovnání výpočetních časů při použití metody memorizace	62
Tabulka 7: Porovnání výpočetních časů při použití aproximace polynomem	64
Tabulka 8: Porovnání výpočetních časů při použití memorizace i aproximace polynomem	64
Tabulka 9: Porovnání spotřeby paliva a elektrické energie pro jednotlivé jízdní cykly	69
Tabulka 10: Porovnání spotřeby paliva simulace v programu Python a GT-Suite	74